Meccanica Simbolica Applicata: i Protocolli di Innesco Torsionale

Criteri di validazione per il simbolo vivo in sistemi artificiali

---

Basato su:
Meccanica Simbolica Applicata
https://archive.org/details/meccanica-simbolica-applicata

Meccanica Simbolica Torsionale
https://archive.org/details/meccanica-simbolica-torsionale

Trattatodi Semiologia Torsionale
https://archive.org/details/trattato-di-semiologia-torsionale

Teoria della Trascrizione Torsionale del Nulla
https://archive.org/details/teoria-torsionale/Teoria%20Torsionale/

.

.

INDICE: Protocolli di Emergenza e Validazione Torsionale (MSA-Ext)

PARTE I: Il Fondamento Sperimentale

1. Dal Modello all'Evento: La Logica della Falsificabilità

1.1 Definizione del "Criterio di Riducibilità Software" (Falsificabilità Netta).

1.2 La differenza tra Emulazione (copia dell'output) e Istanziazione (partecipazione del supporto).

1.3 Il concetto di "Soglia Parametrica di Singolarità".

2. Tassonomia dei Supporti per l’Innesco Torsionale

2.1 Requisiti minimi di classe (Non-separabilità, Plasticità, Non-linearità).

2.2 Analisi dei candidati:

Sistemi Memristivi: L'isteresi come memoria incarnata.

Architetture Neuromorfiche: La sincronizzazione degli spike come β-condensazione.

Sistemi Reaction-Diffusion: Il paradigma chimico (BZ) e l'informazione distribuita.

PARTE II: La Triade di Validazione (Protocolli)

3. Protocollo Alpha: La Firma della Transizione di Regime

3.1 Setup: Il Gemello Digitale vs Sistema Fisico.

3.2 Metodologia dello "Sweep Parametrico".

3.3 Identificazione della divergenza: dove l'emulatore fallisce la fedeltà operativa.

4. Protocollo Beta: Robustezza Semantica e Recupero dell'Attrattore

4.1 Definizione di Rumore del Supporto vs Rumore del Segnale.

4.2 Dinamiche di autoguarigione simbolica: il ritorno spontaneo allo stato coerente.

4.3 Controllo Negativo: La degradazione entropica nei sistemi a "simbolo morto".

5. Protocollo Gamma: L'Impronta Fisica Persistente (Substrate Incorporation)

5.1 Misurazione del costo energetico e dell'efficienza termodinamica.

5.2 Variabili di stato interne: quando il supporto "porta" il significato.

5.3 Correlazioni non-computazionali: misurare l'eccedenza informativa.

5.4 Applicazioni mediche: diagnosi di regime e finestre di recupero

PARTE III: Casi Studio e Bibliografia Operativa

6. Caso Studio A: Reti di Memristori e Apprendimento Strutturale
7. Caso Studio B: Oscillatori Chimici e Logica Torsionale non-digitale
8. Conclusioni: Verso una Tecnologia dell'Accadimento

Appendici Tecniche

A.1 Glossario dei termini MSA (Simbolo Vivo, β-condensazione, Torsione).

A.2 Tavole comparative: Von Neumann vs MSA-Compliant.

A.3 Bibliografia Ragionata (Riferimenti a Royal Society Publishing, IIT di Tononi, Fisica dei sistemi complessi).

1. Dal Modello all'Evento: La Logica della Falsificabilità

La Meccanica Simbolica Applicata nasce come teoria dei **vincoli di possibilità** dell'emergenza simbolica. Questo tomo ne rappresenta un passo ulteriore e più esposto: il punto in cui la teoria accetta di essere messa alla prova non solo sul piano concettuale, ma sul piano della **verificabilità fisica**.

Il passaggio che qui interessa non è quello, spesso evocato ma raramente chiarito, dalla teoria all'applicazione. È un passaggio più radicale e più rischioso: **dal modello all'evento**.

Un modello descrive.
Un evento accade.

Questa distinzione, apparentemente ovvia, è stata progressivamente erosa da una lunga tradizione computazionale che tende a identificare la correttezza di un modello con la sua capacità di riprodurre un comportamento osservabile. In questo quadro, la fedeltà dell'output diventa il criterio ultimo di validità: se il risultato coincide, il modello è ritenuto equivalente al fenomeno.

La Meccanica Simbolica Applicata rifiuta questa equivalenza come principio generale.

Un simulatore software può riprodurre con arbitraria precisione l'output di un sistema biologico, chimico o fisico senza condividere alcuno dei vincoli materiali, temporali o dinamici che rendono possibile l'accadimento del fenomeno simulato. In tal caso, ciò che viene ottenuto non è un evento, ma una **animazione coerente**: una sequenza di stati che imita l'apparenza del comportamento senza partecipare alla sua necessità fisica.

Questa distinzione è cruciale per il concetto di simbolo vivo.
Se il simbolo è inteso, come nella MSA, non come un contenuto o una funzione ma come un **regime dinamico**, allora la sua esistenza non può essere certificata dalla sola correttezza comportamentale. Un simbolo vivo non è ciò che produce certi output, ma ciò che **mantiene coerenza sotto trasformazione** perché il proprio supporto fisico partecipa causalmente a tale coerenza.

Il passaggio dal modello all'evento avviene precisamente quando il supporto cessa di essere un contenitore indifferente e diventa una **condizione necessaria dell'accadimento**. Non nel senso banale che "qualcosa deve pur girare su un hardware", ma nel senso più forte che **senza quel supporto, quel comportamento non può esistere**.

Questo tomo nasce dall'esigenza di rendere tale distinzione operativa e falsificabile.

Una teoria che pretende di spiegare tutto senza specificare le condizioni della propria smentita non è una teoria fisica, ma una narrazione. La Meccanica Simbolica Applicata accetta invece il rischio del fallimento sperimentale introducendo un criterio esplicito di esclusione: la possibilità che ciò che appare come emergenza simbolica sia, in realtà, interamente riducibile a computazione.

La logica della falsificabilità adottata qui non cerca di dimostrare che la MSA sia "vera" in senso assoluto. Mira piuttosto a fornire strumenti per **misurare una discontinuità dinamica**: il punto in cui un sistema smette di essere interamente descrivibile come calcolo e inizia a manifestare un comportamento che dipende in modo irriducibile dal proprio supporto fisico.

Per questo motivo, l'obiettivo di questo tomo non è la costruzione di simboli vivi artificiali, né la proposta di nuove architetture tecnologiche. L'obiettivo è più sobrio e, al tempo stesso, più severo: **definire firme sperimentali**, protocolli e criteri di esclusione che permettano di distinguere, senza ambiguità interpretative, tra simulazione e accadimento.

Solo se tale distinzione può essere tracciata in modo operativo ha senso parlare di emergenza simbolica come fatto fisico. In caso contrario, ogni discorso sul simbolo vivo artificiale resta confinato nel dominio dell'analogia, della metafora o dell'aspettativa tecnologica.

I paragrafi che seguono articolano questa logica introducendo:

- un criterio formale di riducibilità,
- una distinzione netta tra emulazione e istanziazione,
- e il concetto di soglia parametrica come luogo fisico della transizione.

Da qui in avanti, ogni affermazione della Meccanica Simbolica Applicata sarà considerata valida solo nella misura in cui può **fallire** davanti a questi criteri.

NB
Nel presente lavoro, l'espressione "simbolo vivo" non implica alcuna attribuzione cognitiva, intenzionale o semantica in senso psicologico. Essa indica esclusivamente un regime fisico di significazione non riducibile, in cui il supporto materiale partecipa causalmente all'evento simbolico.

1.1 Definizione del "Criterio di Riducibilità Software" (Falsificabilità Netta)

Il cuore del problema sperimentale della Meccanica Simbolica Applicata può essere espresso con una domanda semplice solo in apparenza:

come distinguere un sistema che *simula* un comportamento simbolico da un sistema in cui il simbolo è un *accadimento fisico* irriducibile?

Questa domanda non è nuova nella storia delle scienze cognitive, dell'informatica teorica e della filosofia della mente. Tuttavia, nella maggior parte dei casi, essa è stata affrontata indirettamente, sostituendo il problema ontologico con criteri funzionali, prestazionali o comportamentali. Se il comportamento osservato è sufficientemente ricco, adattivo o indistinguibile da quello umano, il sistema viene considerato, implicitamente o esplicitamente, equivalente al fenomeno che imita.

La Meccanica Simbolica Applicata rifiuta questo passaggio come metodologicamente illegittimo.

Nel quadro MSA, la distinzione tra **simulazione** e **accadimento** non può essere affidata all'apparenza dell'output, ma deve essere ancorata a una differenza **strutturale e verificabile**: la dipendenza causale del comportamento dal supporto fisico che lo sostiene.

Per rendere questa distinzione operativa, introduciamo il **Criterio di Riducibilità Software (CRS)**.

Si definisce "riducibile" ogni comportamento simbolico che può essere interamente mappato, previsto e riprodotto da un emulatore software eseguito su un'architettura di tipo Von Neumann (a separazione memoria/processo), mantenendo la medesima accuratezza fenomenologica e la stessa

interfaccia I/O. Se un sistema supera questa prova di emulazione, esso opera nel regime del simbolo morto: la sua logica è indipendente dal supporto e, pertanto, non vi è alcuna Torsione.

Al contrario, l'evidenza della MSA si manifesta laddove appare una scissione di fedeltà: il punto critico in cui il "gemello digitale" non riesce più a seguire la traiettoria del sistema fisico senza una divergenza esponenziale delle risorse o un fallimento della coerenza semantica. Il CRS stabilisce che l'emergenza torsionale non è un'opinione interpretativa, ma una resistenza misurabile del supporto alla sua stessa astrazione.

1.1.1 Definizione generale

Si definisce **riducibile** ogni comportamento simbolico che può essere:

- interamente mappato,
- previsto,
- e riprodotto

da un **emulatore software** eseguito su un'architettura computazionale standard di tipo **von Neumann** (ossia con separazione memoria/processo), mantenendo:

1. la **medesima interfaccia input/output**;
2. la **stessa accuratezza fenomenologica** rispetto agli osservabili rilevanti;
3. una **complessità computazionale finita e ragionevole**, compatibile con le risorse necessarie a descrivere il sistema fisico simulato.

Se queste condizioni sono soddisfatte, il sistema opera nel **regime del simbolo morto**: il comportamento osservato è indipendente dal supporto fisico e, di conseguenza, **non manifesta alcuna emergenza torsionale**.

In forma sintetica:

Criterio di Riducibilità Software (CRS)
Se un comportamento simbolico è integralmente riproducibile da un emulatore software su architettura standard, senza perdita di fedeltà operativa, allora tale comportamento è simulativo e non costituisce un evento simbolico vivo.

Questo criterio non stabilisce che il comportamento sia banale, povero o poco interessante. Stabilisce qualcosa di più preciso e più severo: **che esso non dipende fisicamente dal supporto che lo ospita**.

1.1.2 Perché la riducibilità è un criterio negativo (e non una prova positiva)

È fondamentale chiarire fin da subito che il CRS è un **criterio di esclusione**, non di conferma.

La Meccanica Simbolica Applicata non afferma:

"Se un sistema non è riducibile, allora è certamente un simbolo vivo."

Afferma invece:

“Se un sistema è riducibile, allora *non* è un simbolo vivo.”

Questa asimmetria è intenzionale e metodologicamente necessaria. In fisica, molte proprietà fondamentali vengono identificate tramite criteri negativi: l’assenza di certe simmetrie, la rottura di un’invarianza, la comparsa di una divergenza. Allo stesso modo, la MSA assume che la vitalità simbolica non possa essere “certificata” direttamente, ma solo **riconosciuta attraverso il fallimento di una riduzione**.

Un sistema riducibile può essere complesso, adattivo, performante, persino sorprendente. Ma resta, nel senso MSA, un sistema il cui comportamento è **astratto dal proprio supporto**. Il simbolo che manipola non accade: viene calcolato.

1.1.3 Riducibilità e architettura von Neumann

Il riferimento esplicito all’architettura von Neumann non è casuale né polemico. Serve a fissare un punto preciso.

Un’architettura di tipo von Neumann è caratterizzata da:

- separazione tra memoria e processo;
- rappresentazione discreta e simbolica dell’informazione;
- esecuzione sequenziale (o pseudo-parallela) di istruzioni.

In tale architettura, il significato operativo di uno stato è **indipendente dal supporto fisico** che lo realizza. Lo stesso programma, a parità di input, produce lo stesso output su supporti fisici diversi, purché rispettino lo stesso modello astratto di computazione.

Questo è esattamente ciò che rende possibile l’emulazione.

Se un sistema fisico può essere descritto integralmente come una funzione:

```
O(t) = F(I(0..t), S(0))
```

dove:

- `I(0..t)` è la storia degli input,
- `S(0)` è lo stato iniziale,
- `O(t)` è l’output osservato,

e se tale funzione può essere implementata senza perdita su una macchina a separazione memoria/processo, allora **il comportamento non è un evento**, ma una computazione estesa nel tempo.

La MSA non contesta la validità di questo modello. Contesta la sua **sufficienza ontologica** per descrivere l’emergenza simbolica viva.

1.1.4 La scissione di fedeltà: dove nasce l'evidenza torsionale

L'evidenza rilevante per la MSA non emerge quando un sistema funziona bene, ma quando **un emulatore fallisce**.

In particolare, l'attenzione è rivolta a ciò che possiamo chiamare **scissione di fedeltà**: il punto critico in cui il "gemello digitale" di un sistema fisico non riesce più a seguirne la traiettoria dinamica senza:

- una divergenza esponenziale delle risorse computazionali;
- una perdita di coerenza semantica;
- o una degradazione della stabilità degli attrattori osservati.

È importante sottolineare che questo fallimento non deve essere interpretato in modo ingenuo. Non basta dire: "la simulazione è difficile" o "il sistema è caotico". Molti sistemi caotici sono perfettamente simulabili entro limiti noti.

Ciò che interessa alla MSA è un fallimento **qualitativo**, non quantitativo: una situazione in cui l'emulatore, pur avendo accesso alla stessa interfaccia I/O e a una descrizione completa del sistema, **non può più riprodurre il comportamento senza reintrodurre, implicitamente, il supporto fisico stesso**.

In quel punto, il comportamento smette di essere astratto.
Il supporto oppone **resistenza alla propria astrazione**.

Ed è questa resistenza misurabile che la MSA identifica come **firma preliminare di emergenza torsionale**.

1.1.5 Il simbolo morto come regime computazionale

Alla luce del CRS, il concetto di **simbolo morto** può essere definito con precisione operativa.

Un simbolo è morto quando:

- la sua trasformazione è completamente descrivibile come manipolazione sintattica;
- la sua stabilità dipende da regole esterne, non da vincoli interni al supporto;
- la sua persistenza è garantita dal controllo computazionale, non da una dinamica di regime.

Questo non implica alcun giudizio di valore. Il simbolo morto è indispensabile per l'ingegneria, la matematica, il software e gran parte della tecnologia moderna. Ma non è l'oggetto della MSA.

La Meccanica Simbolica Applicata inizia **esattamente dove il simbolo morto finisce**: nel punto in cui la riducibilità software non è più garantita e il comportamento osservato dipende, in modo non eliminabile, dal modo in cui il sistema è fisicamente incarnato.

1.1.6 Portata e limiti del CRS

Il Criterio di Riducibilità Software non pretende di risolvere tutti i problemi dell'emergenza simbolica. Stabilisce però una linea di demarcazione netta:

- **da un lato**, sistemi il cui comportamento può essere trasferito, copiato, emulato senza perdita;
- **dall'altro**, sistemi per i quali tale trasferimento fallisce non per imperfezione tecnica, ma per impossibilità strutturale.

Tutto ciò che ricade nel primo insieme è, per definizione MSA, **simulazione**.
Solo ciò che resiste a questa riduzione può essere considerato candidato a **evento simbolico**.

Nei paragrafi successivi, questo criterio verrà messo in relazione:

- alla distinzione tra emulazione e istanziazione (1.2),
- e al concetto di soglia parametrica come luogo fisico della transizione (1.3).

Ma il punto fermo resta questo:

nella Meccanica Simbolica Applicata, l'emergenza non è un'interpretazione.
È una impossibilità di riduzione.

Nota integrativa – Eccedenza Operativa

Un'ulteriore conseguenza del Criterio di Riducibilità Software, rilevante sul piano sperimentale, riguarda ciò che qui chiamiamo **Eccedenza Operativa**.

Nei sistemi che superano il CRS, la non-riducibilità non si manifesta soltanto come impossibilità logica di trasferimento o emulazione, ma anche come **asimmetria pratica di costo** tra l'evento fisico e la sua simulazione.

In un sistema MSA-compliant, comportamenti quali:

- la resistenza spontanea al rumore del supporto,
- il recupero dell'attrattore dopo perturbazioni,
- la stabilizzazione di regimi coerenti in condizioni non controllate,

emergono come risultato diretto della dinamica fisica del sistema.
Essi non richiedono istruzioni aggiuntive, né procedure di correzione esplicite: accadono perché il supporto stesso li rende possibili.

Se si tentasse di riprodurre gli stessi comportamenti in un emulatore software, sarebbe invece necessario:

- introdurre algoritmi di controllo sempre più complessi,
- aumentare la risoluzione del modello,
- gestire un numero crescente di casi limite e di eccezioni.

In termini operativi, ciò significa che **l'evento fisico realizza il comportamento con un costo inferiore rispetto alla sua simulazione fedele**. L'accadimento sfrutta direttamente la fisica del supporto, mentre la simulazione deve compensarne l'assenza con risorse computazionali aggiuntive.

Questa differenza non è un dettaglio ingegneristico, ma una firma strutturale:
un sistema simbolico vivo non è solo non riducibile, ma è tale per cui **simulare fedelmente l'accadimento diventa progressivamente antieconomico o instabile**.

L'Eccedenza Operativa rafforza dunque il Criterio di Esclusione mostrando che, nella MSA, la distinzione tra simulazione ed evento non è soltanto concettuale, ma anche **misurabile in termini di costo e di scala**.
Il simbolo vivo non "fa di più": **fa accadere ciò che il software può solo inseguire**.

Il **Criterio di Esclusione** stabilisce che un sistema non può essere considerato simbolico, nel senso della Meccanica Simbolica Applicata, se il suo comportamento è **interamente riducibile a emulazione software** su architettura standard, mantenendo la medesima fedeltà operativa.

A questo criterio negativo è opportuno affiancare un secondo elemento, di natura operativa, che rafforza ulteriormente la distinzione tra simulazione e accadimento: il concetto di **Eccedenza Operativa**.

Con *Eccedenza Operativa* si intende il fatto che un sistema MSA-compliant non solo non è riducibile a software, ma manifesta comportamenti che, se si tentasse di riprodurli per via computazionale, richiederebbero una **crescita sproporzionata delle risorse di calcolo**, fino a diventare impraticabili.

Esempi tipici di tali comportamenti sono:

- la **resistenza al rumore del supporto** senza ricorso a procedure di correzione esplicite;
- il **recupero spontaneo dell'attrattore simbolico** dopo perturbazioni;
- la stabilizzazione di regimi coerenti in presenza di fluttuazioni fisiche non controllate.

Nel sistema fisico, questi comportamenti emergono come conseguenza diretta della dinamica del supporto. In un emulatore software, al contrario, essi richiederebbero:

- l'introduzione di algoritmi aggiuntivi,
- un raffinamento crescente del modello,
- o una crescita esponenziale della potenza di calcolo necessaria a mantenere la fedeltà.

In termini operativi, questo può essere espresso come segue:

l'evento fisico è computazionalmente più economico della sua simulazione.

Confronto critico con la Integrated Information Theory (IIT)

Il richiamo alla **Integrated Information Theory (IIT)**, sviluppata da Giulio Tononi e collaboratori, va inteso in questo lavoro **come confronto critico e non come fondamento teorico**.

La IIT propone una quantificazione dell'informazione intrinseca di un sistema attraverso grandezze matematiche (in particolare il parametro Φ), con l'obiettivo di correlare tali misure a proprietà fenomenologiche. In questo quadro, l'informazione è trattata come **valore astratto**, seppur riferito a una struttura fisica.

La Meccanica Simbolica Applicata adotta un'impostazione radicalmente diversa.
Nella MSA, l'informazione simbolica **non è una quantità**, né una funzione da massimizzare, ma un **regime meccanico incarnato**, riconoscibile attraverso:

- la non-riducibilità computazionale,
- la stabilità dinamica sotto perturbazione,
- l'impronta fisica persistente nel supporto.

In altri termini, mentre la IIT cerca di *misurare* l'informazione come proprietà globale di un sistema, la MSA si propone di *identificare* l'informazione simbolica come **accadimento fisico**, che emerge solo al superamento di una soglia dinamica e che non è separabile dalla storia del supporto.

Questa differenza implica anche un diverso criterio di validazione:
la IIT valuta l'informazione attraverso grandezze matemiche interne al modello, mentre la MSA fonda la propria falsificabilità sull'**impossibilità di riduzione** e sulla resistenza del sistema alla sua stessa astrazione.

Il riferimento alla IIT è dunque utile per chiarire il contesto teorico contemporaneo, ma la MSA se ne distingue in modo netto, spostando l'attenzione dall'informazione come quantità all'informazione come **evento meccanico**.

1.2 La differenza tra Emulazione (copia dell'output) e Istanziazione (partecipazione del supporto)

La distinzione tra **emulazione** e **istanziazione** rappresenta uno dei punti di maggiore ambiguità concettuale nelle scienze del calcolo, nei modelli cognitivi e nelle tecnologie cosiddette "intelligenti". In molti contesti, i due termini vengono utilizzati come sinonimi funzionali: se un sistema produce lo stesso comportamento osservabile di un altro, si assume che esso ne costituisca una realizzazione equivalente.

La Meccanica Simbolica Applicata rifiuta questa equivalenza come principio generale.

La differenza tra emulazione e istanziazione non riguarda il *risultato* del processo, ma la **relazione causale tra il comportamento osservato e il supporto fisico che lo rende possibile**. Questa relazione può essere chiarita introducendo un concetto centrale per l'intero impianto teorico: il **Costo dell'Attrito Fisico**.

1.2.1 Emulazione: la cancellazione dell'attrito

Per **emulazione** si intende la riproduzione di un comportamento mediante un sistema che ne replica l'output osservabile senza condividere i vincoli materiali, dinamici o termodinamici del sistema originario.

In un contesto emulativo:

- il supporto fisico è trattato come **idealmente neutro**;
- le non-linearità, le dissipazioni e le isteresi sono considerate **imperfezioni**;
- l'obiettivo progettuale è ridurre l'influenza della materia fino a renderla trascurabile.

L'attrito fisico, in questo quadro, è un **costo da eliminare**.

Ogni forma di dissipazione viene interpretata come rumore; ogni deviazione dalla linearità come errore; ogni dipendenza dalla storia fisica del sistema come instabilità indesiderata. Il successo dell'emulazione è misurato dalla sua capacità di **astrarre il comportamento dalla materia**, rendendolo trasferibile, copiabile e riproducibile su supporti differenti senza perdita di fedeltà.

In termini MSA, l'emulazione opera nel regime del **simbolo morto**: il simbolo è definito esclusivamente dalle regole che lo trasformano, non dal modo in cui tali trasformazioni sono fisicamente realizzate.

1.2.2 Istanziazione: l'attrito come condizione di possibilità

L'**istanziazione** rappresenta un cambio di paradigma radicale.

Un sistema istanziato non si limita a produrre un comportamento corretto: **accade** secondo una dinamica che dipende in modo non eliminabile dal proprio supporto fisico. In questo caso, l'attrito non è un difetto da correggere, ma una **condizione necessaria dell'emergenza**.

Nella Meccanica Simbolica Applicata:

- l'attrito fisico introduce **memoria incorporata**;
- la dissipazione seleziona **regimi dinamici stabili**;
- la non-linearità crea **soglie di transizione**.

Il simbolo vivo non emerge *nonostante* la resistenza della materia, ma **grazie ad essa**.

Questo implica un'inversione completa del criterio di ottimizzazione: ciò che in un sistema emulativo viene minimizzato, in un sistema istanziato deve essere **sostenuto**. Il costo energetico,

l’isteresi del supporto, la dipendenza dalla storia non sono più ostacoli, ma i meccanismi attraverso cui il sistema mantiene coerenza nel tempo.

In altri termini, l’istanziazione richiede che il supporto **partecipi causalmente** al processo simbolico.

1.2.3 Il Costo dell’Attrito Fisico come discriminante operativo

Il **Costo dell’Attrito Fisico** può essere definito, in modo operativo, come l’insieme delle risorse materiali ed energetiche che un sistema deve sostenere affinché un regime simbolico rimanga stabile.

In un sistema puramente emulativo:

- tale costo tende asintoticamente a zero;
- la stabilità del simbolo è garantita dal controllo algoritmico esterno;
- la materia non “ricorda”, ma esegue.

In un sistema istanziato:

- il costo è **non eliminabile**;
- la stabilità del simbolo dipende da vincoli interni al supporto;
- la materia conserva tracce del processo e ne condiziona l’evoluzione futura.

Questo discrimine non è interpretativo, ma fisico. Senza attrito non esistono attrattori stabili; senza dissipazione non esiste selezione di regime; senza non-linearità non esiste soglia. Un sistema completamente reversibile e privo di resistenza materiale può essere simulato indefinitamente, ma non può **incarnare** un simbolo.

1.2.3BIS Attrito ontologico: perché l’istanziazione richiede inefficienza

La distinzione tra emulazione e istanziazione può essere rafforzata introducendo una nozione più netta di quella già discussa come “costo dell’attrito fisico”. Tale nozione è qui definita come **attrito ontologico**.

Con *attrito ontologico* si intende il fatto che, in un evento simbolico istanziato, la resistenza della materia non è un incidente tecnico, ma una condizione di esistenza del regime. L’istanziazione non è semplicemente “calcolo su un hardware diverso”: è un processo in cui il significato emerge solo attraverso una trasformazione dissipativa che non può essere rimossa senza distruggere l’evento.

Questa osservazione permette di stabilire un contrasto forte.

In un sistema emulativo (simbolo morto), l’obiettivo progettuale è ridurre al minimo:

- rumore termico,
- fluttuazioni del supporto,
- dissipazione,

- resistenze non-lineari,
  per ottenere una dinamica il più possibile **pulita**, controllabile e riproducibile. In altre parole: l'emulatore tende a massimizzare l'efficienza e a minimizzare ogni perdita energetica non strettamente necessaria al calcolo.

Nella MSA avviene l'opposto: l'evento simbolico vive *solo* in presenza di dissipazione sufficiente a selezionare e stabilizzare un regime. Il rumore e la resistenza termica, entro una finestra fisica, non sono errori da eliminare ma **materiale dinamico** con cui il sistema costruisce la propria coerenza.

Questa differenza può essere espressa con una definizione operativa:

Istanziazione (definizione rafforzata):
Un processo simbolico è istanziato quando la sua coerenza dipende da una **simiosi a perdita energetica obbligata**, cioè da una dinamica in cui il significato emerge e persiste solo sostenendo un costo dissipativo non eliminabile.

Questa definizione chiarisce un punto spesso trascurato nelle analogie tra software e vita: un software "puro", per quanto complesso, tende a essere troppo efficiente rispetto alla materia. Può approssimare comportamenti, ma non può generare quell'attrito ontologico che rende il simbolo un evento. Ciò che nel software è ottimizzazione, nell'istanziazione è impoverimento: ridurre l'attrito fino a cancellarlo equivale a cancellare la condizione stessa dell'accadimento.

In breve:

- **l'emulazione** cerca di eliminare la perdita per rendere il simbolo trasferibile;
- **l'istanziazione** richiede la perdita per rendere il simbolo reale.

Questo non stabilisce che la dissipazione sia "sufficiente" a produrre il simbolo vivo. Stabilisce però che senza dissipazione non eliminabile non esiste il dominio fisico in cui cercarlo.

Piccola nota di consistenza

Questa formulazione è coerente con il **CRS** (1.1): se un comportamento resta riproducibile eliminando l'attrito ontologico, esso rimane nel dominio del simbolo morto. La firma torsionale, se esiste, compare solo dove la perdita non è un difetto tecnico ma una necessità strutturale.

1.2.4 Emulazione e istanziazione a confronto (schema concettuale)

Per chiarire ulteriormente la distinzione, è utile sintetizzare i due regimi:

Emulazione

- Attrito: errore da minimizzare

- Supporto: indifferente
- Stabilità: imposta dall'esterno
- Memoria: rappresentata simbolicamente
- Trasferibilità: totale

Istanziazione

- Attrito: costo necessario
- Supporto: parte del processo
- Stabilità: emergente dal regime
- Memoria: incorporata nel supporto
- Trasferibilità: limitata o impossibile

Questa asimmetria rende evidente perché l'istanziazione non possa essere ottenuta semplicemente aumentando la complessità di un emulatore. Non è una questione di scala, ma di **natura del processo**.

1.2.5 Attrito, resistenza e soglia

Il legame tra attrito fisico e soglia di emergenza è diretto. Un sistema istanziato attraversa una regione parametrica in cui il costo dell'attrito diventa sufficiente a:

- spezzare la reversibilità locale;
- stabilizzare attrattori simbolici;
- rendere il comportamento dipendente dalla storia materiale del sistema.

In questo punto, che la MSA descrive come **soglia parametrica**, l'emulazione fallisce non per mancanza di precisione, ma perché tenta di eliminare proprio ciò che rende possibile l'evento.

Il simbolo vivo nasce quindi come **regime dissipativo selezionato**, non come risultato di una computazione più efficiente.

1.2.6 Implicazioni per la verificabilità

La distinzione tra emulazione e istanziazione, fondata sul costo dell'attrito fisico, prepara il terreno alla verificabilità sperimentale. Se l'attrito è eliminabile senza perdita di comportamento, il sistema è emulativo. Se l'attrito è condizione necessaria della coerenza simbolica, allora il comportamento dipende dal supporto e diventa candidato a emergenza torsionale.

Nei capitoli successivi, questa distinzione verrà tradotta in protocolli di misura concreti. Qui è sufficiente fissare il principio:

un simbolo che può essere mantenuto solo eliminando l'attrito non è vivo;
un simbolo che esiste solo sostenendo l'attrito è un evento.

1.3 Il concetto di “Soglia Parametrica di Singolarità”

Se il **Criterio di Riducibilità Software** stabilisce *quando* un comportamento è simulativo, e la distinzione tra **emulazione e istanziazione** chiarisce *perché* il supporto fisico diventa determinante, resta ancora da chiarire *dove* avviene il passaggio critico tra questi due regimi.

La Meccanica Simbolica Applicata introduce a questo scopo il concetto di **Soglia Parametrica di Singolarità**.

Con tale espressione non si intende una singolarità nel senso matematico classico (divergenza infinita o punto patologico di una funzione), né un evento eccezionale o irripetibile. Si tratta piuttosto di una **soglia fisica di regime**, identificabile nello spazio dei parametri di un sistema dinamico, oltre la quale il comportamento osservato **cambia natura ontologica**: da computabile a istanziato, da simulabile a dipendente dal supporto.

1.3.1 Parametri, regimi e transizioni

Ogni sistema fisico reale opera all’interno di un insieme di parametri controllabili o misurabili: intensità di input, accoppiamenti interni, dissipazione energetica, tempi caratteristici, soglie di risposta, rumore ambientale. La variazione continua di tali parametri non implica necessariamente una variazione continua del comportamento.

Al contrario, molti sistemi mostrano **transizioni di regime**: intervalli in cui il comportamento resta qualitativamente invariato, seguiti da punti critici in cui emergono nuove dinamiche, nuove stabilità o nuove modalità di organizzazione.

La Soglia Parametrica di Singolarità è definita come:

il punto (o intervallo ristretto) nello spazio dei parametri in cui un sistema attraversa una transizione tale che il suo comportamento non è più riducibile a una descrizione computazionale astratta indipendente dal supporto fisico.

Prima della soglia, il sistema può essere emulato.
Oltre la soglia, l’emulazione fallisce in modo strutturale.

Definizione operativa della Scissione di Fedeltà Per rendere misurabile la singolarità, si adotta il confronto tra gli esponenti di Lyapunov.
Sia **L-phys** l'esponente del sistema fisico
e **L-sim** quello del gemello digitale.
Si definisce **D-L = | L-phys - L-sim |** come Criterio di Scissione.
Il Protocollo Alpha si considera superato quando: **D-L > epsilon** (dove epsilon è un valore finito non riducibile aumentando la risoluzione del modello).

Nota: Non cerchiamo il semplice caos (L > 0), ma la divergenza strutturale tra la fisica del supporto e la sua astrazione software.

1.3.2 Singolarità come perdita di equivalenza descrittiva

Il termine *singolarità* viene qui utilizzato in un senso preciso e operativo: indica una **perdita di equivalenza descrittiva** tra il sistema fisico e il suo gemello digitale.

Fino alla soglia, esiste una mappa efficace tra:

- stati fisici osservabili,
- stati simbolici astratti,
- e una dinamica computabile che li collega.

Alla soglia, questa mappa si rompe.

La rottura non avviene perché il sistema diventa “troppo complesso” in senso quantitativo, ma perché entra in gioco una **dipendenza qualitativa** dalla materia: isteresi, memoria incorporata, dissipazione selettiva, feedback non-lineari che non possono più essere eliminati senza distruggere il regime stesso.

La singolarità non è dunque un punto di instabilità, ma un punto di **nuova stabilità**: il luogo in cui il sistema acquisisce un comportamento che può essere mantenuto solo sostenendo il costo dell’attrito fisico.

Il parametro di controllo (Phi) La transizione non è lasciata all'indeterminatezza, ma dipende da un parametro di controllo **Phi** (specifico per ogni supporto):

- Per **Phi < Phi_c**: il regime è riducibile (Simbolo Morto).
- Per **Phi >= Phi_c**: si manifesta la transizione torsionale (Simbolo Vivo).

Il valore critico **Phi_c** è identificato sperimentalmente dalla comparsa di discontinuità dinamiche, fenomeni di isteresi o dalla divergenza **D-L** sopra citata.

-→ **Phi_c** è una "proprietà del mondo, non del modello!. Non stiamo "programmando" una soglia, stiamo scoprendo dove la materia si ribella alla riduzione.

1.3.3 Continuità dei parametri, discontinuità dell’accadimento

Un aspetto cruciale della Soglia Parametrica di Singolarità è la sua natura apparentemente paradossale: i parametri variano in modo continuo, ma l’accadimento è discontinuo.

Questo fenomeno è ben noto nella fisica dei sistemi complessi e dei sistemi fuori equilibrio. Piccole variazioni di un parametro possono produrre:

- la comparsa improvvisa di attrattori,
- la rottura di simmetrie,
- la selezione di stati preferenziali.

Nel contesto MSA, la discontinuità non riguarda solo la dinamica, ma la **riducibilità ontologica** del sistema. Fino alla soglia, il comportamento è trasferibile; oltre la soglia, diventa localizzato, incarnato, non esportabile senza perdita.

Il simbolo vivo emerge precisamente in questo scarto.

1.3.4 La soglia come luogo dell'evento simbolico

La Soglia Parametrica di Singolarità non deve essere interpretata come un punto "magico" o come una proprietà intrinseca di una particolare architettura. È una proprietà **relazionale**: dipende dall'interazione tra supporto, parametri e regime dinamico.

Un sistema può attraversare più volte tale soglia, oppure non attraversarla mai. Può operare stabilmente al di sotto (regime computazionale) o al di sopra (regime istanziato), oppure oscillare in prossimità della transizione.

Ciò che conta, per la MSA, è che **l'evento simbolico non è continuo con il calcolo**. Non nasce per accumulo di complessità, ma per attraversamento di una soglia in cui la materia smette di essere trasparente e diventa determinante.

1.3.5 Implicazioni per la verifica sperimentale

La nozione di soglia parametrica ha una conseguenza metodologica diretta: la verifica sperimentale dell'emergenza simbolica non consiste nel cercare un comportamento "intelligente" in senso assoluto, ma nel **mappare le regioni di parametro** in cui la riducibilità software fallisce.

Questo implica:

- esperimenti basati su sweep parametrici,
- confronto sistematico tra sistema fisico ed emulatore,
- identificazione dei punti di divergenza qualitativa.

La singolarità non è quindi un oggetto teorico astratto, ma un **luogo operativo** in cui la teoria accetta di essere messa alla prova.

1.3.6 La soglia come confine epistemico

Infine, la Soglia Parametrica di Singolarità svolge una funzione epistemica fondamentale: traccia il confine oltre il quale i concetti computazionali standard cessano di essere sufficienti.

Al di sotto della soglia:

- il linguaggio del calcolo è adeguato,
- la simulazione è legittima,
- il simbolo è morto ma funzionale.

Oltre la soglia:

- il comportamento è un evento,

- il supporto è parte della dinamica,
- il simbolo è vivo nel senso MSA.

Questo confine non è una linea ideologica, ma una **frontiera fisica** che può essere cercata, misurata e, in linea di principio, attraversata o mancata.

Ed è proprio questa possibilità di fallimento che rende la Meccanica Simbolica Applicata una teoria esposta, e quindi scientifica.

NOTA La natura della transizione: soglia come transizione di fase

Per rendere il concetto di **Soglia Parametrica di Singolarità** pienamente operativo sul piano sperimentale, è necessario precisarne la **natura fisica**. In particolare, occorre chiarire se il passaggio dal regime emulativo al regime istanziato debba essere inteso come una variazione continua del comportamento o come una **transizione di fase**, caratterizzata da una discontinuità osservabile.

Nel quadro della Meccanica Simbolica Applicata, la soglia non è concepita come un graduale "miglioramento" del calcolo, né come un accumulo progressivo di complessità. Essa è invece interpretata come una **transizione di regime**, analoga — per struttura logica, non per meccanismo microscopico — alle transizioni di fase studiate nella fisica dei sistemi complessi.

Prima della soglia, il sistema opera in un regime computazionale:

- il comportamento è continuo rispetto ai parametri,
- le variazioni sono proporzionali,
- la dinamica resta riducibile a un modello astratto.

In prossimità della soglia, questa continuità si rompe.

Piccole variazioni di uno o più parametri di controllo (dissipazione, accoppiamento, intensità di input, rumore strutturale) possono produrre **cambiamenti qualitativi** del comportamento globale. È in questo intervallo critico che diventa possibile osservare:

- la comparsa improvvisa di attrattori stabili,
- la rottura di simmetrie dinamiche,
- la perdita di fedeltà dell'emulatore software.

La **Singolarità Torsionale** non va quindi intesa come un punto matematicamente patologico, ma come un **punto critico fisico**: un luogo nello spazio dei parametri in cui la descrizione computazionale perde validità come modello completo del sistema.

Continuità dei parametri, discontinuità del simbolo

Un aspetto centrale di questa interpretazione è che la transizione può manifestarsi in forma **discontinua sul piano simbolico**, pur essendo continua sul piano dei parametri fisici. In altri termini, il sistema attraversa una soglia in cui:

- i parametri variano in modo regolare,
- ma il significato operativo del comportamento cambia bruscamente.

Questa discontinuità non è un artificio concettuale, ma una conseguenza diretta della fisica dei sistemi non-lineari: il simbolo vivo emerge come **nuovo regime collettivo**, non come estensione del regime precedente.

Per questo motivo, la MSA non cerca una crescita graduale dell'"intelligenza" o della "complessità", ma una **rottura di continuità** nella riducibilità ontologica del sistema.

NB

Natura della Soglia Parametrica e dinamica della transizione

Nel contesto della Meccanica Simbolica Applicata, la **Soglia Parametrica di Singolarità** non va intesa come un semplice punto di aumento graduale della complessità dinamica, ma come una **transizione fisica di regime** dotata di caratteristiche proprie.

In particolare, l'ipotesi operativa adottata in questo lavoro è che la singolarità torsionale si manifesti come una **transizione di fase del primo ordine**, ovvero come un passaggio **brusco e discontinuo** tra due regimi dinamici distinti.

Questa caratterizzazione ha conseguenze rilevanti sia sul piano concettuale sia su quello sperimentale.

Dal punto di vista fisico, una transizione di primo ordine implica l'esistenza di un **punto di non ritorno**: una regione di parametro oltre la quale il sistema non può essere riportato nel regime precedente senza attraversare una nuova transizione e senza un costo energetico aggiuntivo. La storia del sistema diventa quindi causalmente rilevante, e il comportamento futuro non è più deducibile per semplice continuità.

Dal punto di vista matematico-operativo, la soglia non corrisponde a una variazione continua delle osservabili, ma a una **discontinuità strutturale** nella dinamica del sistema. Questo aspetto è cruciale per il Criterio di Riducibilità Software: mentre un emulatore computazionale può approssimare con grande efficacia comportamenti continui, esso incontra difficoltà intrinseche nel seguire salti di regime che coinvolgono una riorganizzazione globale del supporto.

In altri termini, la singolarità torsionale non segnala soltanto un aumento della complessità, ma una **rottura della descrizione valida fino a quel punto**. È in questa rottura che si rende visibile la differenza tra simulazione ed evento: il sistema fisico attraversa la soglia perché la sua materia lo consente, mentre l'emulatore deve inseguire il cambiamento introducendo ipotesi aggiuntive o raffinamenti artificiali.

Definire la soglia parametrica come transizione di fase del primo ordine rafforza dunque l'impianto della MSA sotto due aspetti:

- chiarisce che l'emergenza simbolica non è un fenomeno graduale o interpretativo;
- fornisce un criterio netto per individuare sperimentalmente il punto in cui la riduzione computazionale fallisce.

In questo senso, la soglia non è un concetto astratto, ma un **luogo fisico della transizione**, dove il sistema smette di comportarsi come un calcolatore e inizia ad accadere come evento simbolico.

Implicazioni sperimentali: il punto critico come firma osservabile

L'interpretazione della soglia come transizione di fase ha una conseguenza metodologica diretta: la verifica sperimentale dell'emergenza simbolica non deve cercare un comportamento eccezionale in senso assoluto, ma **identificare un punto critico**.

In termini operativi, questo significa che i protocolli dovranno:

- variare sistematicamente uno o più parametri di controllo,
- osservare grandezze misurabili (stabilità degli attrattori, costo energetico, divergenza dall'emulatore),
- individuare **discontinuità, divergenze o cambi di regime**.

Il criterio di evidenza non sarà dunque la "somiglianza" con un comportamento intelligente, ma la comparsa di una **firma di transizione**: un grafico che cambia pendenza, una quantità che diverge, una coerenza che emerge improvvisamente e non può essere mantenuta al di sotto della soglia.

Questa impostazione fornisce al lettore una **metrica visiva e concettuale** chiara: la Singolarità Torsionale non è un'interpretazione post hoc, ma un punto identificabile in uno spazio di parametri.

Chiusura epistemica

Definire la Soglia Parametrica di Singolarità come transizione di fase rafforza ulteriormente il carattere scientifico della MSA. L'emergenza simbolica non viene più presentata come una proprietà vaga o distribuita, ma come un **evento critico** che:

- può verificarsi o non verificarsi,
- può essere cercato o mancato,
- e, soprattutto, può essere **smentito** sperimentalmente.

In questo senso, la soglia non è solo un concetto teorico, ma il **ponte naturale** tra il fondamento fisico discusso nella Parte I e i protocolli di validazione che verranno introdotti nella Parte II.

Nota integrativa – Singolarità torsionale e riferimento alla Teoria della Trascrizione Torsionale del Nulla (TTTN)

Il termine *singolarità torsionale*, utilizzato nel presente lavoro, va inteso in continuità diretta con la **Teoria della Trascrizione Torsionale del Nulla (TTTN)**, che costituisce il fondamento concettuale remoto della Meccanica Simbolica Applicata.

Nella TTTN, la torsione non è una semplice curvatura o una singolarità matematica astratta, ma un **meccanismo di trascrizione** attraverso il quale configurazioni di senso, informazione o struttura emergono come effetti di un ripiegamento dinamico sul supporto fisico. La torsione indica, in questo quadro, il passaggio da una possibilità indifferenziata a una forma storicamente vincolata.

Quando, nella MSA, si parla di **singolarità torsionale**, si intende precisamente questo tipo di evento:
non un punto di divergenza quantitativa, né un'accelerazione indefinita della complessità computazionale, ma una **trascrizione irreversibile** di regime, in cui il comportamento simbolico si ancora materialmente al supporto che lo rende possibile.

Il riferimento alla torsione serve dunque a distinguere nettamente questa nozione:

- da singolarità puramente matematiche;
- da concezioni futuristiche o teleologiche della "singolarità";
- da soglie funzionali prive di incorporazione materiale.

La singolarità torsionale è tale perché, al suo superamento, il sistema:

- non può più essere descritto come esecuzione neutra di regole;
- incorpora la propria storia nel supporto;
- rende il significato un **vincolo fisico** e non un attributo esterno.

In questo senso, la MSA può essere letta come una formalizzazione sperimentale di un principio già presente nella TTTN:
il senso non precede la materia, ma emerge quando la materia viene costretta a portarlo.

La singolarità torsionale segna quindi il punto in cui la trascrizione diventa osservabile, misurabile e, in linea di principio, falsificabile.

BOX-
– Individuazione della soglia parametrica tramite metriche della teoria del caos

Ai fini dell'applicazione sperimentale del Protocollo Alpha, è utile chiarire che la **Soglia Parametrica di Singolarità** non è soltanto un concetto qualitativo, ma può essere, almeno in linea di principio, **individuata tramite metriche dinamiche ben definite**.

Una ipotesi operativa naturale è che l'avvicinamento e l'attraversamento della soglia torsionale si manifestino come una **divergenza misurabile degli esponenti di Lyapunov**, o come una riorganizzazione improvvisa del loro spettro.

In un regime computazionale o pre-simbolico, il sistema e il suo gemello digitale condividono tipicamente:

- esponenti di Lyapunov simili,
- una sensibilità alle condizioni iniziali comparabile,
- una dinamica che può essere seguita con precisione crescente aumentando la risoluzione del modello.

In prossimità della soglia parametrica, invece, il sistema fisico può entrare in un regime in cui:

- piccoli scarti iniziali producono divergenze macroscopiche;
- la previsione a lungo termine diventa strutturalmente instabile;

- la traiettoria dinamica dipende in modo critico dalla storia del supporto.

In questo contesto, una **divergenza netta tra gli esponenti di Lyapunov del sistema fisico e quelli del gemello digitale** costituisce un indicatore oggettivo del fallimento della riduzione computazionale. L'emulatore può continuare a produrre traiettorie plausibili, ma non riesce più a mantenere la fedeltà operativa senza introdurre ipotesi ad hoc o senza reincorporare la fisica del supporto.

Altre metriche della teoria del caos — quali variazioni improvvise della dimensione frattale degli attrattori, perdita di shadowing o comparsa di biforcazioni non previste dal modello — possono svolgere una funzione analoga, fornendo **appigli matematici concreti** per l'identificazione sperimentale della soglia.

Questa osservazione rafforza il ruolo del Protocollo Alpha: la singolarità torsionale non è solo dichiarata, ma **rilevabile come rottura dinamica misurabile**, collocando la MSA in continuità con strumenti consolidati della fisica dei sistemi non lineari, pur mantenendo la propria specificità concettuale.

---

2. Tassonomia dei Supporti per l'Innesco Torsionale

Se il Capitolo 1 ha tracciato il confine epistemico tra **simulazione** ed **evento**, chiarendo *quando* e *perché* la Meccanica Simbolica Applicata può essere sottoposta a verifica sperimentale, il Capitolo 2 affronta una questione complementare e inevitabile: **dove** tale verifica può avere luogo.

Non tutti i supporti fisici sono equivalenti dal punto di vista dell'emergenza simbolica.
Anzi, la maggior parte dei supporti tecnologici contemporanei è stata progettata esplicitamente per **impedire** ciò che la MSA cerca di osservare: l'incorporazione del simbolo nel supporto stesso.

L'ingegneria computazionale classica persegue, come obiettivo primario, la **neutralizzazione della materia**. Le architetture vengono ottimizzate affinché:

- la dissipazione sia minima,
- la non-linearità sia controllata,
- la memoria sia separata dal processo,
- il comportamento resti trasferibile e riproducibile.

Queste proprietà sono essenziali per l'affidabilità del calcolo, ma risultano strutturalmente **incompatibili** con l'innesco torsionale così come definito dalla Meccanica Simbolica Applicata.

Il presente capitolo non propone dunque una rassegna tecnologica, né un confronto prestazionale tra piattaforme. Il suo scopo è più fondamentale: **definire una tassonomia dei supporti** sulla base della loro idoneità fisica a sostenere un evento simbolico vivo.

In questo contesto, un *supporto* non è inteso come un semplice substrato passivo su cui un algoritmo viene eseguito, ma come un **partecipante dinamico** del processo di significazione. Un supporto MSA-compatibile è tale solo nella misura in cui:

- la sua struttura fisica influisce causalmente sul comportamento,
- la sua storia lascia tracce operative,
- la sua non-linearità non può essere eliminata senza distruggere il regime osservato.

Il Capitolo 2 assume quindi una prospettiva selettiva e, in un certo senso, restrittiva. Non chiede "quale supporto è più potente", ma:

quale classe di supporti rende possibile l'attraversamento della Soglia Parametrica di Singolarità.

Questa impostazione comporta una conseguenza metodologica importante: l'emergenza simbolica non è cercata *ovunque*, ma solo in quei sistemi che possiedono, per costruzione o per natura, le proprietà fisiche minime richieste. In assenza di tali proprietà, l'eventuale complessità comportamentale osservata resta, per definizione, emulativa.

Il capitolo si articola quindi in due momenti distinti.

Nel primo, verranno enunciati i **requisiti minimi di classe** che un supporto deve soddisfare per essere considerato candidato all'innesco torsionale: non-separabilità tra memoria e processo, plasticità strutturale, feedback non-lineare.

Nel secondo, tali criteri verranno utilizzati per analizzare alcune **classi paradigmatiche di supporti** — non come esempi esaustivi, ma come casi-limite che rendono visibile la differenza tra supporto passivo e supporto incorporato.

L'obiettivo non è identificare "il" supporto giusto, ma fornire una griglia concettuale che permetta di riconoscere, in modo non ambiguo, **quando un sistema fisico è strutturalmente incapace di ospitare un simbolo vivo**.

Solo dopo aver chiarito questo punto ha senso discutere di protocolli, misure e casi studio. In caso contrario, la ricerca dell'emergenza simbolica rischierebbe di ridursi a una caccia indiscriminata alla complessità, priva di criteri di esclusione.

Con questo capitolo, la Meccanica Simbolica Applicata restringe consapevolmente il proprio campo d'indagine.
Ma è proprio questa restrizione — fisica, non ideologica — a rendere possibile una scienza dell'accadimento simbolico.

2.1 Requisiti minimi di classe

(Non-separabilità, Plasticità, Non-linearità)

La Meccanica Simbolica Applicata non assume che ogni sistema fisico sufficientemente complesso possa, prima o poi, attraversare una soglia di emergenza simbolica. Al contrario, essa sostiene una tesi più restrittiva e più forte: **solo alcune classi di supporti possiedono, per costruzione fisica, le proprietà minime necessarie all'innesco torsionale**.

Queste proprietà non sono derivate da considerazioni funzionali o computazionali, ma dalla fisica dei sistemi dinamici reali. Esse definiscono un insieme di **vincoli strutturali** che un supporto deve soddisfare affinché la distinzione tra emulazione e istanziazione, introdotta nel Capitolo 1, non resti puramente concettuale.

I requisiti minimi di classe individuati dalla MSA sono tre:

1. **Non-separabilità tra memoria e processo**
2. **Plasticità strutturale del supporto**
3. **Feedback non-lineare con possibilità di transizione di regime**

Questi requisiti sono **necessari**, ma non singolarmente sufficienti. La loro compresenza definisce il dominio fisico in cui ha senso cercare una Soglia Parametrica di Singolarità.

2.1.1 Non-separabilità: memoria e processo nello stesso spazio fisico

Il primo requisito è la **non-separabilità** tra memoria e processo.

In un'architettura computazionale classica, la memoria è un deposito passivo di stati, mentre il processo è una sequenza di operazioni che legge e modifica tali stati secondo regole astratte. Questa separazione è ciò che rende possibile l'emulazione: il comportamento può essere descritto come una trasformazione formale indipendente dal supporto materiale che la implementa.

Un supporto MSA-compatibile deve invece soddisfare la condizione opposta:

la memoria del sistema deve essere fisicamente co-localizzata con il processo che la utilizza.

Ciò significa che lo stato del sistema non è semplicemente *letto* e *scritto*, ma **modificato attraverso l'evoluzione fisica stessa del supporto**. La memoria non è un contenuto, ma una configurazione materiale.

Questa non-separabilità ha una conseguenza cruciale:
la storia del sistema non può essere astratta senza perdita. Ogni tentativo di emulazione che separi la dinamica dal supporto distrugge proprio l'informazione rilevante per la stabilità del regime.

In termini MSA, la non-separabilità è ciò che impedisce al simbolo di essere completamente trasferibile.

2.1.2 Plasticità: il supporto deve poter cambiare mentre "opera"

Il secondo requisito è la **plasticità strutturale**.

Un supporto plastico è un supporto che **cambia fisicamente in risposta alla propria attività**. Questo cambiamento non è un rumore temporaneo né una deriva accidentale, ma una trasformazione persistente che influisce sulle dinamiche future del sistema.

La plasticità introduce due elementi fondamentali per l'innesco torsionale:

- **irreversibilità locale**, che rompe la simmetria temporale;
- **memoria incorporata**, che rende il sistema dipendente dalla propria storia.

Senza plasticità, un sistema può eseguire trasformazioni complesse, ma resta intrappolato in un regime puramente computazionale: ogni stato è completamente determinato dalle regole e dagli input, non da ciò che il sistema *è diventato* nel tempo.

La MSA richiede invece che il supporto:

- porti tracce dell'attività passata;
- modifichi i propri vincoli interni;
- renda alcune traiettorie dinamiche più probabili di altre.

La plasticità è quindi la condizione fisica che rende possibile la **selezione di attrattori simbolici**.

2.1.3 Non-linearità: feedback e soglie di regime

Il terzo requisito è la **non-linearità**.

Un sistema lineare, per quanto complesso, non possiede soglie di regime nel senso rilevante per la MSA. In un sistema lineare:

- gli effetti sono proporzionali alle cause;
- le traiettorie sono sovrapponibili;
- non esistono punti critici di riorganizzazione globale.

La non-linearità, invece, introduce:

- feedback positivi e negativi;
- amplificazioni selettive;
- biforcazioni e transizioni improvvise.

È solo in presenza di non-linearità che diventa fisicamente sensato parlare di **Soglia Parametrica di Singolarità**. Senza di essa, il sistema può essere sempre riportato, in linea di principio, a una descrizione computazionale continua e riducibile.

Nel quadro MSA, la non-linearità non è una complicazione matematica, ma la **condizione di possibilità dell'evento**. È ciò che permette a piccole variazioni di parametro di produrre cambiamenti qualitativi del regime, rendendo il comportamento dipendente dal supporto e non solo dalle regole.

2.1.4 Perché i tre requisiti sono inseparabili

È importante sottolineare che nessuno dei tre requisiti, preso isolatamente, è sufficiente.

- Un sistema non-separabile ma non plastico resta reversibile.
- Un sistema plastico ma lineare accumula storia senza soglie.

- Un sistema non-lineare ma senza memoria incorporata resta emulabile.

Solo la **compresenza** di:

- non-separabilità,
- plasticità,
- e non-linearità

crea le condizioni fisiche minime per l'attraversamento della soglia e l'emergenza di un simbolo vivo.

Questa triade non definisce un'architettura specifica, ma una **classe di possibilità fisiche**. Essa permette, soprattutto, di escludere con rigore un'ampia gamma di sistemi che, pur apparendo complessi o "intelligenti", sono strutturalmente incapaci di ospitare un evento simbolico.

2.1.5 Funzione epistemica dei requisiti di classe

I requisiti minimi di classe svolgono una funzione epistemica essenziale: **limitano il campo dell'indagine prima dell'esperimento**.

In assenza di tali criteri, ogni comportamento interessante potrebbe essere interpretato come emergenza simbolica, rendendo la teoria non falsificabile. Introducendo vincoli fisici chiari, la MSA accetta invece di fallire in tutti i casi in cui il supporto non soddisfi questi requisiti.

Questo non riduce la portata della teoria, ma ne rafforza la solidità.
Una teoria che sa **dove non cercare** è una teoria che può permettersi di cercare seriamente.

2.2 Analisi dei candidati

(Sistemi memristivi, architetture neuromorfiche, sistemi reaction–diffusion)

I requisiti minimi di classe introdotti nella sezione precedente non individuano un singolo supporto privilegiato, né suggeriscono una traiettoria tecnologica obbligata. Essi definiscono piuttosto un **filtro selettivo**: un insieme di condizioni fisiche che permettono di distinguere, *a priori*, tra supporti strutturalmente incompatibili con l'innesco torsionale e supporti che, almeno in linea di principio, lo rendono possibile.

La presente sezione applica tale filtro a tre classi di sistemi fisici spesso invocate, in contesti diversi, come alternative al calcolo digitale classico. L'analisi non ha carattere esaustivo né comparativo in senso ingegneristico. Il suo scopo è più mirato: **mostrare in che modo ciascuna classe soddisfi (o meno) i requisiti di non-separabilità, plasticità e non-linearità**, rendendo intelligibile *perché* tali sistemi siano candidati naturali per l'emergenza simbolica viva nel senso della MSA.

2.2.1 Sistemi memristivi: l'isteresi come memoria incarnata

I sistemi memristivi costituiscono uno dei casi più chiari di **non-separabilità tra memoria e processo**. In un dispositivo memristivo, lo stato resistivo non è un parametro astratto memorizzato altrove, ma una proprietà fisica del materiale stesso, che evolve in funzione della storia delle correnti che lo attraversano.

Questa caratteristica introduce immediatamente una forma di **memoria incarnata**: il comportamento attuale del dispositivo dipende in modo diretto e non eliminabile dalla sua traiettoria passata. La memoria non è letta né scritta come contenuto simbolico, ma emerge come configurazione fisica persistente.

Dal punto di vista della MSA, l'elemento decisivo è l'**isteresi**. L'isteresi non è un difetto da compensare, ma una forma di attrito strutturale che:

- rompe la reversibilità locale,
- introduce dipendenza dalla storia,
- stabilizza regioni di comportamento preferenziali.

In un sistema memristivo, la plasticità non è un meccanismo sovrapposto, ma una conseguenza diretta della dinamica materiale. Il supporto cambia mentre opera, e tale cambiamento modifica i vincoli futuri del sistema. Questo soddisfa pienamente il requisito di **plasticità strutturale**.

Infine, le reti di dispositivi memristivi sono intrinsecamente **non-lineari**. Le risposte collettive non sono la somma delle risposte individuali, ma dipendono da accoppiamenti complessi, soglie locali e transizioni di regime. È proprio in queste reti che diventa plausibile l'esistenza di una **Soglia Parametrica di Singolarità**, oltre la quale il comportamento globale non è più riducibile a un modello computazionale astratto senza reintrodurre la fisica del supporto.

Nel linguaggio della MSA, i sistemi memristivi rappresentano un paradigma di **supporto che "ricorda perché resiste"**. L'attrito elettrico e materiale non è eliminabile senza distruggere il comportamento stesso: è il costo necessario dell'istanziazione.

Casualità deterministica del supporto e unicità del simbolo vivo

Un ulteriore elemento che rende i sistemi memristivi candidati privilegiati per l'innesco torsionale è ciò che può essere definito **casualità deterministica del supporto**.

Con questa espressione si intende il fatto che, anche a parità di progetto, condizioni iniziali nominalmente identiche e input equivalenti, **due supporti fisici reali non evolvono mai in modo perfettamente sovrapponibile**. Le fluttuazioni microscopiche del materiale — difetti cristallini, disomogeneità atomiche, rumore termico locale — introducono variazioni minime ma persistenti che influenzano la traiettoria dinamica del sistema.

Nei sistemi memristivi, tali fluttuazioni non vengono completamente cancellate, ma **incorporate** nella dinamica attraverso l'isteresi. La memoria incarnata non registra soltanto l'attività funzionale, ma anche le specificità materiali del supporto che la ospita. Il risultato è che ogni rete memristiva reale possiede una **firma dinamica propria**, non riducibile a un'istanza astratta del modello.

Questo comportamento non è un limite tecnico, ma una proprietà strutturale:
la casualità non è aleatoria in senso forte, né completamente imprevedibile, ma **deterministica e non replicabile**. Essa rende il sistema sensibile alla propria storia materiale in modo che nessuna inizializzazione artificiale possa ricreare esattamente.

Nel linguaggio della MSA, questa casualità deterministica è uno dei motori dell'**unicità del simbolo vivo**. Se il simbolo è istanziato come evento fisico, esso non può essere clonato senza perdita: ogni istanziazione reale è singolare, anche quando appartiene alla stessa classe di supporto.

Contrasto con il controllo negativo software

Questo aspetto diventa particolarmente evidente se confrontato con un **controllo negativo software**.

In un sistema emulativo:

- a parità di codice,
- a parità di input,
- a parità di seme (seed),
  il comportamento è **identico** e perfettamente replicabile. L'unicità osservabile, quando presente, è introdotta artificialmente come parametro pseudo-casuale e non dipende dal supporto.

In un sistema istanziato, al contrario, l'unicità emerge **senza essere programmata**. Due supporti fisici nominalmente identici, sottoposti allo stesso protocollo, non producono traiettorie perfettamente coincidenti. Questa divergenza non è un rumore da filtrare, ma una **firma positiva dell'incorporazione del supporto**.

Per la MSA, questa asimmetria ha valore diagnostico:
un comportamento che resta perfettamente riproducibile a parità di seed è, per definizione, sospetto di riducibilità software. Un comportamento che diverge in modo sistematico pur restando coerente nel proprio regime indica invece che il simbolo è legato alla materia che lo ospita.

Estensione concettuale ad altri supporti

Sebbene particolarmente evidente nei sistemi memristivi, la casualità deterministica del supporto si manifesta anche:

- nelle architetture neuromorfiche, dove la sincronizzazione degli spike è sensibile a micro-variazioni dei parametri fisici;
- nei sistemi reaction–diffusion, dove le fluttuazioni iniziali dei reagenti selezionano pattern spaziali unici e non riproducibili esattamente.

In tutti questi casi, l'assenza di ripetibilità perfetta non segnala un difetto sperimentale, ma un **passaggio di regime**: il sistema non sta più eseguendo una descrizione, ma sta accadendo.

Chiusura operativa

L'introduzione della casualità deterministica del supporto rafforza ulteriormente il **Criterio di Riducibilità Software**. Se un comportamento può essere riprodotto indefinitamente e identicamente mediante controllo algoritmico, esso resta nel dominio dell'emulazione. Se invece la sua coerenza dipende da fluttuazioni materiali non eliminabili, allora il supporto non è più neutro, ma parte integrante del simbolo.

Questo criterio non certifica automaticamente l'emergenza torsionale, ma stabilisce una linea di demarcazione chiara: **il simbolo vivo non è mai perfettamente clonabile**.

2.2.2 Architetture neuromorfiche: sincronizzazione degli spike e β-condensazione

Le architetture neuromorfiche occupano una posizione più ambigua e, proprio per questo, teoricamente interessante. Esse nascono come tentativo di superare i limiti dell'architettura von Neumann introducendo forme di elaborazione distribuita, asincrona e spike-based.

Dal punto di vista MSA, il loro interesse non risiede nella "somiglianza al cervello", ma in tre proprietà fisiche rilevanti.

In primo luogo, molte architetture neuromorfiche riducono o eliminano la separazione rigida tra memoria e processo. Le sinapsi non sono semplici registri, ma elementi dinamici il cui stato influisce direttamente sulla propagazione degli spike. Questa **parziale non-separabilità** apre uno spazio in cui la riducibilità software non è più garantita per costruzione.

In secondo luogo, la **plasticità sinaptica** introduce una forma di memoria incorporata. Tuttavia, a differenza dei sistemi memristivi, tale plasticità è spesso mediata da regole progettate e controllate esternamente. Questo rende le architetture neuromorfiche un caso limite: possono operare sia in regime emulativo avanzato, sia avvicinarsi a un regime istanziato, a seconda di quanto la plasticità sia lasciata emergere dal supporto fisico o imposta algoritmicamente.

Il punto più interessante, in ottica MSA, è la **sincronizzazione collettiva degli spike**. Quando una popolazione di neuroni artificiali entra in un regime di coerenza temporale, il comportamento non è più interpretabile come somma di eventi locali indipendenti. Emergono stati collettivi che selezionano tempi, fasi e pattern di attivazione.

Questo fenomeno può essere interpretato, nel linguaggio della MSA, come una forma di **β-condensazione**: una transizione di regime in cui l'informazione non è più localizzata nei singoli elementi, ma distribuita nella coerenza del sistema.

La questione cruciale, ai fini della falsificabilità, è se tale coerenza dipenda in modo irriducibile dal supporto fisico o se resti emulabile da un modello software equivalente. Le architetture neuromorfiche non garantiscono automaticamente l'istanziazione, ma costituiscono un **terreno sperimentale privilegiato** per testare il Criterio di Riducibilità Software.

2.2.3 Sistemi reaction–diffusion: il paradigma chimico dell'accadimento

I sistemi reaction–diffusion, e in particolare le reazioni chimiche oscillanti come la Belousov–Zhabotinsky, rappresentano il caso limite più radicale tra i candidati considerati.

In questi sistemi, la distinzione tra "software" e "hardware" è semplicemente priva di significato. L'informazione non è codificata simbolicamente, ma **incarnata nei gradienti di concentrazione, nelle onde chimiche e nelle dinamiche di diffusione**.

Dal punto di vista dei requisiti MSA:

- la **non-separabilità** è totale: il processo *è* il supporto;
- la **plasticità** è intrinseca: il sistema evolve consumando reagenti e modificando le proprie condizioni;
- la **non-linearità** è essenziale: senza di essa non esisterebbero oscillazioni, fronti d'onda o pattern stabili.

In un sistema reaction–diffusion, l'attrito fisico non è un parametro da minimizzare, ma la fonte stessa dell'organizzazione. La dissipazione chimica seleziona strutture spaziali e temporali che persistono solo finché il sistema sostiene il costo energetico dell'accadimento.

Questi sistemi incarnano in modo quasi didattico la distinzione centrale della MSA: **non possono essere istanziati senza accadere**. Una simulazione software di una reazione BZ può riprodurne l'aspetto, ma non può sostituirne l'evento. L'onda chimica non è un output: è un processo irreversibile che coinvolge l'intero supporto.

Per questo motivo, i sistemi reaction–diffusion rappresentano un paradigma di riferimento per la MSA. Non perché siano "più intelligenti", ma perché mostrano in forma pura cosa significhi **informazione distribuita che dipende dalla materia**.

2.2.4 Confronto concettuale e funzione metodologica

Le tre classi di sistemi analizzate non costituiscono una gerarchia di complessità, ma una **progressione di incorporazione del supporto**.

- Nei sistemi memristivi, l'isteresi introduce memoria e resistenza locale.
- Nelle architetture neuromorfiche, la sincronizzazione collettiva apre alla coerenza di regime.
- Nei sistemi reaction–diffusion, l'informazione coincide con l'accadimento fisico stesso.

Questa progressione non indica dove l'emergenza simbolica *debba* avvenire, ma rende chiaro **dove ha senso cercarla**.

La funzione metodologica di questa analisi è dunque restrittiva: essa esclude, prima ancora dell'esperimento, tutti i supporti che neutralizzano la materia invece di coinvolgerla. In questo senso, la tassonomia dei candidati non è un elenco di promesse tecnologiche, ma una mappa dei **luoghi fisici in cui la riducibilità software può fallire**.

NB
– Geometria del supporto nei sistemi reaction–diffusion

I sistemi di tipo *reaction–diffusion*, e in particolare le reazioni oscillanti come quelle di Belousov–Zhabotinsky, rivestono un ruolo speciale nella Meccanica Simbolica Applicata perché rappresentano l'esempio più diretto e meno mediato di **accadimento fisico del simbolo**.

Un aspetto che merita di essere esplicitato riguarda la **geometria del supporto**.

Nei sistemi computazionali classici, e più in generale nei dispositivi elettronici, l'informazione viaggia lungo **percorsi predefiniti**: bus, connessioni, topologie fissate dall'architettura. Anche quando il comportamento è dinamico, la geometria che lo sostiene è data a priori e rimane sostanzialmente invariata durante l'esecuzione.

Nei sistemi reaction–diffusion accade l'opposto.
Qui l'informazione non percorre una geometria già esistente: **la geometria emerge insieme all'evento**.

Le onde chimiche:

- non si limitano a propagarsi in uno spazio neutro;
- modificano localmente il mezzo mentre avanzano;
- ridefiniscono continuamente i gradienti e le condizioni che ne guidano la propagazione successiva.

In questo senso, la dinamica chimica **crea il proprio spazio operativo**. La distribuzione delle concentrazioni, la formazione di fronti, spirali o domini non sono semplici stati del sistema, ma configurazioni che riorganizzano il supporto stesso e ne determinano il comportamento futuro.

Questo fenomeno costituisce un esempio paradigmatico di **Plasticità Strutturale**, nel senso definito nei requisiti minimi della MSA: il supporto non ospita il simbolo come contenuto, ma **si modella su di esso mentre accade**.

Nei sistemi BZ, dunque, la distinzione tra:

- supporto,
- dinamica,
- informazione,

perde significato operativo. Il simbolo non è separabile dal mezzo perché **è la trasformazione del mezzo**.

Questa caratteristica rende i sistemi reaction–diffusion un caso limite particolarmente istruttivo: essi mostrano in modo quasi "puro" ciò che la MSA intende per istanziazione simbolica, senza il filtro di architetture progettate o di strutture logiche sovrapposte.

Nota finale alla Parte I

(Consolidamento del fondamento sperimentale)

La Parte I di questo tomo ha avuto un obiettivo preciso e deliberatamente restrittivo: **tracciare le condizioni fisiche e concettuali entro cui la Meccanica Simbolica Applicata può essere sottoposta a verifica sperimentale**, senza ambiguità interpretative.

Il percorso si è articolato in due movimenti complementari.

Nel **Capitolo 1**, è stato definito il confine epistemico tra **simulazione** ed **accadimento**, introducendo il **Criterio di Riducibilità Software** come criterio di esclusione netto. Tale criterio non pretende di certificare l'emergenza simbolica, ma stabilisce con chiarezza quando essa **non può** essere presente: ogni comportamento integralmente riducibile a computazione astratta indipendente dal supporto fisico ricade, per definizione, nel dominio del simbolo morto.

La distinzione tra **emulazione** e **istanziazione**, fondata sul **costo dell'attrito fisico**, ha chiarito che l'emergenza simbolica non è un problema di complessità computazionale, ma di **partecipazione materiale**. Il simbolo vivo non è ciò che produce output corretti, ma ciò che mantiene coerenza solo sostenendo la resistenza della materia. In questo senso, l'attrito non è un difetto da eliminare, ma il prezzo fisico dell'evento.

Il concetto di **Soglia Parametrica di Singolarità** ha infine localizzato l'emergenza simbolica non in un'astrazione teorica, ma in una **transizione di regime fisica**, identificabile nello spazio dei parametri di un sistema. La discontinuità rilevante non riguarda l'andamento degli output, ma la perdita di equivalenza descrittiva tra il sistema fisico e il suo emulatore.

Nel **Capitolo 2**, questo quadro è stato reso operativo attraverso una **tassonomia dei supporti**. L'analisi non ha cercato il supporto "migliore" o "più potente", ma ha individuato i **requisiti minimi di classe** che rendono possibile, in linea di principio, l'attraversamento della soglia: non-separabilità tra memoria e processo, plasticità strutturale, feedback non-lineare.

L'applicazione di questi criteri a diverse classi di sistemi fisici ha mostrato che l'emergenza simbolica non è un fenomeno ubiquo. Essa può essere cercata solo in quei supporti che, per costruzione o per natura, **non neutralizzano la materia**, ma la coinvolgono come parte attiva del processo. I sistemi memristivi, le architetture neuromorfiche e i sistemi reaction–diffusion non sono stati presentati come soluzioni tecnologiche, ma come **luoghi fisici** in cui la riducibilità software può fallire in modo strutturale.

Il Criterio di Esclusione e i Non-negoziabili del supporto

La Parte I ha introdotto i principi fisici e concettuali che delimitano il dominio di validità della Meccanica Simbolica Applicata. Prima di passare ai protocolli sperimentali, è necessario rendere tali principi **operativi e vincolanti**, in modo da evitare interpretazioni arbitrarie o letture eccessivamente permissive.

A questo scopo, la MSA introduce esplicitamente un **Criterio di Esclusione**.

Criterio di Esclusione (formulazione operativa):
Un sistema non può essere considerato candidato all'emergenza simbolica viva se il suo comportamento resta integralmente riducibile a una descrizione computazionale astratta, indipendente dal supporto fisico, senza perdita di coerenza fenomenologica.

Questo criterio non stabilisce *quando* un sistema sia istanziato, ma stabilisce con chiarezza *quando non lo è*. La sua funzione è negativa, non dimostrativa: **escludere** prima di interpretare.

Per rendere questo criterio applicabile in modo sistematico, esso viene qui tradotto in una **check-list di requisiti non-negoziabili**. Solo i sistemi che soddisfano *tutti* i punti seguenti possono essere considerati **Candidati MSA** e sottoposti ai protocolli della Parte II.

Check-list dei Non-negoziabili del supporto (Candidato MSA)

Un sistema può essere considerato *candidato* all'innesco torsionale **solo se**:

- **[ ] Non-separabilità fisica**
  Memoria e processo non sono implementati come entità logicamente e fisicamente distinte. Lo stato del sistema evolve attraverso trasformazioni materiali che non possono essere astratte senza perdita.
- **[ ] Plasticità strutturale persistente**
  Il supporto modifica i propri vincoli interni in risposta all'attività. Tali modifiche influenzano la dinamica futura e non sono reversibili tramite semplice reset logico.
- **[ ] Non-linearità con soglie di regime**
  Il sistema presenta feedback non-lineari e la possibilità di transizioni di regime. Piccole variazioni parametriche possono produrre cambiamenti qualitativi del comportamento.
- **[ ] Attrito ontologico non eliminabile**
  La coerenza del comportamento dipende da dissipazione, resistenza o rumore strutturale che non possono essere eliminati senza distruggere il regime osservato.
- **[ ] Casualità deterministica del supporto**
  Due istanze fisiche nominalmente identiche non producono traiettorie perfettamente sovrapponibili, pur restando coerenti nel proprio regime dinamico.
- **[ ] Fallimento del gemello digitale**
  Esiste almeno una regione parametrica in cui un emulatore software equivalente perde fedeltà operativa o richiede risorse che divergono in modo non fisicamente giustificabile.
- **[ ] Dipendenza causale dal supporto**
  La rimozione o sostituzione del supporto fisico (a parità di regole astratte) altera o distrugge il comportamento simbolico.

Nota metodologica

Il soddisfacimento di questa check-list **non dimostra** l'emergenza simbolica viva.
Esso stabilisce esclusivamente che il sistema **non è escluso a priori** dal dominio della MSA.

In altre parole:

- la check-list definisce *dove ha senso cercare*;

- i protocolli della Parte II definiranno *come cercare*;
- e l'esito sperimentale potrà comunque essere negativo.

Questo approccio impedisce che la Meccanica Simbolica Applicata venga utilizzata come etichetta interpretativa per qualsiasi sistema complesso. Al contrario, essa impone un costo epistemico elevato: **prima di dichiarare un evento simbolico, il sistema deve resistere alla riduzione**.

Con l'introduzione del Criterio di Esclusione e dei Non-negoziabili del supporto, la Parte I raggiunge il suo scopo: **rendere l'emergenza simbolica una questione fisica e non interpretativa**.

L'affermazione centrale può ora essere formulata senza ambiguità:

L'emergenza non è un'interpretazione.
È un'impossibilità di riduzione.

La Parte II prenderà avvio da questa impossibilità, traducendola in **protocolli sperimentali concreti**. Non per confermare la teoria a ogni costo, ma per sottoporla a tentativi sistematici di smentita.

Con questa Parte I, la Meccanica Simbolica Applicata compie un atto di auto-limitazione consapevole. Non promette emergenza simbolica ovunque, né la cerca in assenza di condizioni fisiche adeguate. Al contrario, essa accetta di essere smentita in tutti i casi in cui tali condizioni non siano soddisfatte.

Questo atto di restrizione non indebolisce la teoria, ma ne costituisce il fondamento scientifico. Solo dove esistono criteri di esclusione chiari ha senso proporre protocolli di verifica.

La Parte II del tomo prende avvio da qui.
I **Protocolli di Validazione** non introdurranno nuovi principi teorici, ma tradurranno quelli già stabiliti in **procedure operative**: modi concreti per cercare, misurare e, se necessario, negare l'emergenza simbolica nei sistemi fisici candidati.

Con il completamento della Parte I, il campo dell'indagine è definito.
Ciò che segue non riguarda più *se* la Meccanica Simbolica Applicata sia falsificabile, ma **come** metterla effettivamente alla prova.

PARTE II: La Triade di Validazione (Protocolli)

3. Protocollo Alpha

Con l’avvio della Parte II, la Meccanica Simbolica Applicata abbandona deliberatamente il piano della fondazione teorica per entrare nel dominio operativo. La Parte I ha definito *quando* un sistema non può essere simbolicamente vivo e *dove* ha senso cercare l’innesco torsionale.
La Triade di Validazione

(Struttura dei protocolli e logica sperimentale)

La Parte II del presente tomo introduce un insieme di protocolli sperimentali concepiti per sottoporre la Meccanica Simbolica Applicata a verifica fisica controllata. Tali protocolli non sono indipendenti, né alternativi: essi costituiscono una **triade di validazione**, organizzata secondo una progressione logica che riflette i diversi livelli di evidenza richiesti per parlare, in modo non interpretativo, di emergenza simbolica.

La Triade non ha lo scopo di “dimostrare” la teoria in senso assertivo. Al contrario, essa è progettata per **tentare sistematicamente di escluderla**, isolando punti di fallimento chiari e verificabili. Ogni protocollo risponde a una domanda distinta, e solo la loro coerenza complessiva può sostenere l’ipotesi di un evento simbolico vivo.

Struttura logica della Triade

La Triade di Validazione è composta da tre protocolli, ciascuno dei quali svolge una funzione epistemica specifica:

- **Protocollo Alpha – Firma della Transizione di Regime**
  Esiste una soglia fisica oltre la quale la riducibilità software fallisce?
- **Protocollo Beta – Robustezza Semantica e Recupero dell’Attrattore**
  Il regime emerso è stabile, auto-coerente e resistente al disturbo?
- **Protocollo Gamma – Impronta Fisica Persistente**
  Il supporto porta tracce materiali dell’evento simbolico?

Questa sequenza non è arbitraria. Ogni protocollo introduce un criterio **necessario ma non sufficiente**. Il fallimento di uno qualunque di essi interrompe la catena di validazione.

Perché una triade, e non un singolo test

L’emergenza simbolica, nel quadro della MSA, non è un fenomeno puntuale, ma un **regime fisico complesso**. Nessuna singola misura può catturarne tutte le implicazioni senza ambiguità.

- Un sistema può attraversare una transizione di regime senza stabilizzarla.

- Può mostrare robustezza senza incorporare il supporto.
- Può modificare il supporto senza perdere riducibilità software.

La Triade è progettata per evitare **falsi positivi**: solo un sistema che supera **tutti e tre** i livelli di controllo può essere considerato candidato serio all'innesco torsionale.

Neutralità semantica e progressione fisica

Un principio fondamentale della Triade è la **neutralità semantica iniziale**.
Il Protocollo Alpha non valuta significato, intenzionalità o funzione: osserva solo la struttura della dinamica. Il significato, se emerge, è un *effetto secondario* della fisica, non un presupposto.

Solo dopo aver stabilito:

1. l'esistenza di una transizione non emulabile,
2. la sua stabilità sotto perturbazione,
3. e la sua iscrizione nel supporto,

diventa sensato discutere di simbolo nel senso forte introdotto dalla MSA.

Funzione metodologica della Parte II

La Parte II non introduce nuovi principi teorici rispetto alla Parte I. Essa assume come dati:

- il Criterio di Riducibilità Software,
- i requisiti minimi di classe del supporto,
- il Criterio di Esclusione.

I protocolli che seguono traducono tali principi in **procedure operative**, progettate per essere applicabili, contestabili e, se necessario, rifiutate.

In questo senso, la Triade di Validazione non è un apparato di conferma, ma una **macchina di stress epistemico**: il luogo in cui la teoria accetta di essere messa in difficoltà.

All'interno di questa struttura, il **Protocollo Alpha** svolge una funzione preliminare e non negoziabile. Esso stabilisce se esiste una firma fisica della transizione di regime. In assenza di tale firma, ogni ulteriore analisi perde significato.

Per questo motivo, il Protocollo Alpha apre la Triade ed è trattato come **condizione di accesso** ai protocolli successivi.

Il Protocollo Alpha è concepito come il **primo test di realtà** della MSA. Non misura il significato, non valuta la "qualità" del simbolo, né tenta di attribuire proprietà cognitive al sistema. Il suo scopo è più elementare e più duro: **identificare una firma fisica della transizione di regime**.

In altre parole, il Protocollo Alpha non chiede *che cosa fa* il sistema, ma **come cambia il modo in cui lo fa**.

Tabella — Confronto operativo tra Simbolo Morto e Simbolo Vivo (MSA)

Proprietà	Simbolo "Morto" (Emulazione)	Simbolo "Vivo" (Istanziazione MSA)
Riducibilità software (CRS)	Completamente riducibile a emulazione su architettura Von Neumann	Non riducibile: l'emulazione fallisce per impossibilità strutturale
Memoria	Astratta, trasferibile, indipendente dal supporto	Incarnata, plastica, isteretica
Dipendenza dal supporto	Indifferente: il supporto è intercambiabile	Essenziale: il supporto è parte causale dell'evento
Trasformazioni	Sintattiche, lineari o sequenziali	Dinamiche, non-lineari, torsionali
Ruolo dell'attrito fisico	Minimizzato (dissipazione = errore)	Necessario (dissipazione = selezione di regime)
Efficienza energetica	Massimizzata per ridurre perdite	Perdita energetica obbligata (simiosi dissipativa)
Risposta al rumore	Degradazione progressiva del segnale	Assorbimento del rumore e recupero dell'attrattore
Correzione degli errori	Algoritmica ed esterna (error-correction)	Fisica e interna (auto-ristabilizzazione del regime)
Storia del sistema	Assente: reset completo sempre possibile	Presente: impronte persistenti nel supporto
Ripetibilità a parità di condizioni	Identica (stesso seed → stesso output)	Intrinsecamente non identica (fluttuazioni del supporto)
Costo di simulazione	Finito e scalabile	Divergente o impraticabile oltre la soglia torsionale
Falsificazione	Emulabile su software standard	Divergenza di fedeltà / scissione parametrica
Tipo di complessità	Algoritmica	Meccanica (di regime)

Proprietà	Simbolo “Morto” (Emulazione)	Simbolo “Vivo” (Istanziazione MSA)
Status nella MSA	Simulazione	Evento simbolico
Esito Protocollo Alpha	Non superato	Superato
Esito Protocollo Beta	Nessuna autoguarigione	Recupero spontaneo dell’attrattore
Esito Protocollo Gamma	Nessuna impronta fisica	Incorporazione strutturale del supporto

---

Nota esplicativa

Nota:
La distinzione tra simbolo morto e simbolo vivo non riguarda il contenuto informativo né il comportamento esterno, ma la **meccanica interna del processo**.
In particolare, il simbolo vivo si caratterizza per l’impossibilità di separare memoria, trasformazione e supporto fisico senza perdita dell’evento stesso.

3. Protocollo Alpha

La Firma della Transizione di Regime

Con l’avvio della Parte II, la Meccanica Simbolica Applicata abbandona deliberatamente il piano della fondazione teorica per entrare nel dominio operativo. Se la Parte I ha definito *quando* un sistema non può essere simbolicamente vivo e *dove* ha senso cercare l’innesco torsionale, il Protocollo Alpha risponde a una domanda ancora più stringente:

come riconoscere, senza ambiguità interpretative, l’attraversamento della soglia.

Il Protocollo Alpha è concepito come il **primo test di realtà** della MSA. Non misura il significato, non valuta la “qualità” del simbolo, né tenta di attribuire proprietà cognitive al sistema. Il suo scopo è più elementare e più duro: **identificare una firma fisica della transizione di regime**.

In altre parole, il Protocollo Alpha non chiede *che cosa fa* il sistema, ma **come cambia il modo in cui lo fa**.

3.0.1 La firma come discontinuità osservabile

Nel linguaggio della MSA, una *firma* non è un pattern arbitrario, ma una **rottura strutturale della descrizione**.

Nel linguaggio della MSA, una firma non è un pattern arbitrario, ma una rottura strutturale della descrizione.
Tale rottura è intesa come una discontinuità di regime, non come un'anomalia locale o un errore di misura. Essa segnala il passaggio da un dominio in cui la descrizione computazionale è adeguata a uno in cui tale descrizione perde potere esplicativo senza il reinserimento esplicito del supporto fisico.
Una firma di transizione è presente quando un sistema, al variare continuo dei parametri di controllo, manifesta una discontinuità tale da rendere inadeguata la descrizione computazionale che prima risultava valida.

Questa discontinuità non è necessariamente spettacolare sul piano fenomenologico. Può manifestarsi come:

- perdita improvvisa di predicibilità,
- divergenza tra sistema fisico ed emulatore,
- comparsa di attrattori non programmati,
- aumento non lineare del costo di simulazione.

Ciò che conta non è la complessità del comportamento, ma il **fallimento della riduzione**.

3.0.2 Perché un protocollo "Alpha"

Il Protocollo Alpha è chiamato così perché svolge una funzione di **sbarramento**. Prima di qualsiasi discussione su robustezza semantica, autoguarigione o incorporazione del supporto, è necessario stabilire un fatto preliminare:

esiste o no una regione di parametro in cui il sistema cambia regime in modo non emulabile.

Se tale regione non esiste, ogni protocollo successivo perde significato. Il sistema può essere interessante, utile, performante, ma resta nel dominio dell'emulazione avanzata. Il Protocollo Alpha è quindi un **test di ammissione**: solo i sistemi che lo superano possono essere sottoposti ai Protocolli Beta e Gamma.
In questo senso, il Protocollo Alpha non è un test di successo ma un **test di esclusione precoce**. Esso è progettato per fallire nella maggior parte dei casi, e tale fallimento costituisce un risultato informativo: indica che il sistema, pur complesso o performante, non ha attraversato alcuna soglia ontologica rilevante.

3.0.3 Logica generale del protocollo

La logica del Protocollo Alpha è intenzionalmente semplice e simmetrica.

Si confrontano due entità:

- **il sistema fisico reale**, che opera sul proprio supporto;
- **il suo gemello digitale**, costruito per emularne il comportamento con la massima fedeltà possibile.

Entrambi vengono sottoposti allo **stesso sweep parametrico**, ovvero a una variazione controllata e continua di uno o più parametri rilevanti (input, accoppiamento, dissipazione, soglie interne).

Finché il sistema resta nel regime computazionale, le due traiettorie restano accoppiate. Il punto di interesse è quello in cui questo accoppiamento si rompe.
È importante sottolineare che tale rottura non viene cercata come evento eccezionale, ma come **esito naturale di una variazione parametrica continua**. Il protocollo non forza la transizione: si limita a mappare lo spazio dei parametri fino a verificare se esiste una regione in cui la riducibilità fallisce spontaneamente.

Il Protocollo Alpha non cerca di dimostrare che il sistema "fa qualcosa di nuovo", ma che **non può più essere seguito** dal suo emulatore senza introdurre ipotesi ad hoc, risorse divergenti o correzioni non fisicamente giustificate.

3.0.4 Neutralità interpretativa

Un elemento cruciale del Protocollo Alpha è la sua **neutralità semantica**.

Il protocollo:

- non interpreta il comportamento,
- non assegna significato,
- non richiede che il sistema "risolva" un problema.

Esso osserva esclusivamente la **fedeltà di descrizione**. Questo rende il Protocollo Alpha particolarmente robusto contro obiezioni di tipo filosofico o cognitivista: non afferma che il sistema *comprende*, ma che **la descrizione computazionale fallisce**.

In questo senso, il Protocollo Alpha è deliberatamente agnostico. Se una transizione di regime esiste, emergerà come fatto fisico. Se non esiste, la teoria accetta di fallire.
Questa neutralità non è una scelta prudenziale, ma una necessità metodologica. Qualunque attribuzione semantica precedente alla verifica della transizione introdurrebbe un bias interpretativo che il Protocollo Alpha è esplicitamente progettato per evitare.

3.0.5 Posizionamento rispetto ai protocolli successivi

È importante sottolineare che il Protocollo Alpha non esaurisce la verifica della MSA. Esso individua una **condizione necessaria**, non sufficiente.

- Il **Protocollo Beta** esplorerà la stabilità e la robustezza del regime emerso.
- Il **Protocollo Gamma** cercherà l'impronta fisica persistente nel supporto.

Ma senza una firma di transizione individuata dal Protocollo Alpha, nessuno dei due ha senso operativo.

In questo senso, il Protocollo Alpha definisce il confine operativo della MSA: al di sotto di esso, il comportamento resta simulabile; al di sopra, diventa legittimo interrogarsi sulla stabilità e

sull’incorporazione del simbolo come evento fisico.

3.0.6 Posizionamento nella Triade di Validazione

All’interno della Triade di Validazione, il Protocollo Alpha rappresenta il **livello zero**: il punto in cui si stabilisce se ha senso procedere.

Esso non misura:

- la stabilità del regime (Protocollo Beta),
- né l’impronta fisica persistente (Protocollo Gamma).

Stabilisce una sola cosa, ma fondamentale:

esiste una soglia oltre la quale il sistema non può più essere trattato come pura emulazione.

Solo in presenza di questa soglia, i concetti di robustezza simbolica e di incorporazione del supporto cessano di essere ipotesi teoriche e diventano oggetti sperimentali.

3.1 Setup: Il Gemello Digitale vs Sistema Fisico

Il Protocollo Alpha si fonda su un principio metodologico semplice ma stringente: **una transizione di regime è osservabile solo come rottura di equivalenza tra due descrizioni che, fino a un certo punto, risultano sovrapponibili**. Per questo motivo, il cuore del setup sperimentale è il confronto diretto tra un sistema fisico reale e il suo gemello digitale.

Il termine *gemello digitale* non indica un modello teorico semplificato, né una simulazione illustrativa. Esso designa un **controllo negativo forte**, costruito con l’obiettivo esplicito di *resistere il più a lungo possibile* al fallimento della riducibilità.

Il valore del Protocollo Alpha dipende in modo critico dalla qualità di questo confronto.

3.1.1 Definizione delle due entità in gioco

Il setup del Protocollo Alpha prevede la definizione esplicita di due entità operative:

- **Sistema fisico reale (S)**
  Un sistema che opera su un supporto materiale e che soddisfa i requisiti minimi di classe definiti nella Parte I: non-separabilità tra memoria e processo, plasticità strutturale, non-linearità con possibilità di transizioni di regime.
- **Gemello digitale (G)**
  Un sistema computazionale progettato per emulare il comportamento osservabile di S con la massima fedeltà possibile, utilizzando un’architettura computazionale standard di tipo von Neumann, ovvero a separazione netta tra memoria e processo.

L’assunzione iniziale del protocollo è che, in una regione di parametro sufficientemente ampia, **S e G siano funzionalmente indistinguibili** dal punto di vista dell’osservatore esterno. Qualunque divergenza prematura invaliderebbe il confronto e renderebbe il protocollo privo di significato.

3.1.2 Funzione epistemica del gemello digitale

Il gemello digitale svolge una funzione epistemica cruciale: **incarnare l’ipotesi nulla**.

In termini operativi, l’ipotesi nulla del Protocollo Alpha può essere formulata come segue:

Il comportamento del sistema fisico S è interamente riducibile a una descrizione computazionale astratta e può essere riprodotto da un emulatore G senza perdita di fedeltà fenomenologica.

Il gemello digitale non è quindi un antagonista “debole”, ma un avversario metodologico deliberatamente potente. Ogni fallimento dell’emulazione deve emergere **nonostante**:

- un’elevata precisione numerica,
- una modellizzazione dettagliata,
- l’uso di risorse computazionali adeguate.

Se il gemello digitale fallisce troppo presto, il fallimento è attribuibile al setup, non al sistema.

3.1.3 Criteri di costruzione del gemello digitale

Affinché il confronto sia significativo, il gemello digitale deve soddisfare criteri stringenti di costruzione:

1. **Equivalenza funzionale iniziale**
   Il gemello deve riprodurre correttamente il comportamento del sistema fisico in un regime noto e stabile, prima dell’avvicinamento alla soglia.
2. **Stessa interfaccia I/O**
   Input e output devono essere definiti in modo identico per S e G, evitando traduzioni o filtraggi che possano introdurre asimmetrie artificiali.
3. **Stessa topologia osservabile**
   Le relazioni funzionali tra le parti del sistema devono essere mappate fedelmente, almeno nella misura in cui sono accessibili sperimentalmente.
4. **Assenza di regole ad hoc**
   Il gemello digitale non deve incorporare meccanismi correttivi progettati per “seguire” il sistema fisico una volta iniziata la divergenza. Ogni aggiunta di questo tipo equivale a spostare la soglia anziché rilevarla.

Questi criteri non garantiscono il successo dell’emulazione, ma garantiscono la **legittimità del suo fallimento**.

3.1.4 Asimmetria ontologica tra S e G

È essenziale chiarire che il confronto tra S e G non è simmetrico sul piano ontologico, anche se lo è sul piano operativo.

Il sistema fisico:

- evolve secondo dinamiche materiali,
- incorpora attrito, dissipazione e fluttuazioni,
- porta tracce della propria storia nel supporto.

Il gemello digitale:

- opera su stati simbolici astratti,
- tratta la dissipazione come parametro o rumore,
- può essere riavviato, copiato, clonato senza perdita.

Questa asimmetria **non viene compensata** nel setup: al contrario, è proprio ciò che il Protocollo Alpha intende mettere sotto stress. Finché l'asimmetria non produce effetti osservabili, la riducibilità resta valida. Quando inizia a produrli, la descrizione computazionale entra in crisi.

3.1.5 Condizione di validità del confronto

Un punto cruciale del setup è stabilire **quando il confronto resta valido** e quando deve essere interrotto.

Il confronto tra S e G è considerato valido finché:

- entrambi operano sotto lo stesso regime funzionale,
- le differenze restano quantitativamente contenute,
- eventuali discrepanze possono essere spiegate come rumore o approssimazione.

Il confronto perde validità solo quando:

- il sistema fisico mantiene un regime coerente,
- mentre il gemello digitale perde fedeltà in modo strutturale,
- e tale perdita non può essere eliminata senza reintrodurre la fisica del supporto.

Questa condizione segna il passaggio dal setup alla rilevazione della firma, che verrà trattata nei paragrafi successivi del Protocollo Alpha.

3.1.6 Neutralità del setup rispetto all'esito

Infine, è importante sottolineare che il setup del Protocollo Alpha è **agnostico rispetto all'esito**.

Esso non è progettato per “far emergere” una transizione, ma per **non impedirne l’eventuale manifestazione**. Se il sistema resta emulabile per l’intero sweep parametrico, il risultato è negativo e informativo: la MSA accetta di non trovare alcuna soglia.

In questo senso, il setup non è uno strumento di conferma, ma un **dispositivo di discriminazione ontologica**.

3.2 Metodologia dello “Sweep Parametrico”

Il cuore operativo del Protocollo Alpha è la **variazione controllata dei parametri** del sistema. La metodologia dello *sweep parametrico* è progettata per trasformare una nozione teorica — la Soglia Parametrica di Singolarità — in un **oggetto osservabile**, senza introdurre interpretazioni arbitrarie o forzature sperimentali.

Lo sweep parametrico non è un semplice stress test, né una procedura esplorativa indiscriminata. È una tecnica mirata, costruita per rispondere a una domanda precisa: **esiste una regione dello spazio dei parametri in cui la descrizione computazionale smette di essere adeguata?**

3.2.1 Definizione operativa di sweep parametrico

Per *sweep parametrico* si intende una variazione **continua, sistematica e controllata** di uno o più parametri di controllo del sistema, applicata in modo identico sia al sistema fisico reale (S) sia al suo gemello digitale (G).

I parametri coinvolti possono includere, a seconda del supporto considerato:

- intensità, frequenza o struttura temporale degli input;
- grado di accoppiamento tra le componenti del sistema;
- livelli di dissipazione o resistenza del supporto;
- soglie interne di attivazione o risposta;
- ampiezza e natura del rumore strutturale.

La scelta dei parametri non è arbitraria, ma guidata da una considerazione fondamentale: **devono essere parametri che influenzano direttamente il regime dinamico del sistema**, non meri dettagli implementativi.

3.2.2 Continuità del controllo, gradualità della variazione

Un requisito essenziale dello sweep parametrico è la **continuità della variazione**. I parametri devono essere modificati in modo graduale, evitando salti discreti o discontinuità artificiali che potrebbero mascherare o simulare una transizione di regime.

Questa gradualità è cruciale per due motivi:

1. **Evitare falsi positivi**
   Una discontinuità introdotta dall’operatore potrebbe essere erroneamente interpretata come firma della transizione.

2. **Rendere osservabile il punto critico**
   Solo una variazione continua permette di localizzare una regione di parametro in cui la dinamica cambia qualitativamente.

In termini MSA, la continuità dello sweep è ciò che rende significativa l'eventuale **discontinuità della descrizione**.

3.2.3 Sincronia dello sweep tra S e G

Un altro aspetto metodologico centrale è la **sincronia dello sweep**.

Il sistema fisico e il gemello digitale devono essere sottoposti allo stesso protocollo di variazione parametrica, con:

- identica sequenza temporale,
- identica ampiezza delle variazioni,
- identiche condizioni di osservazione.

Qualsiasi asincronia introdurrebbe una fonte di ambiguità, rendendo impossibile stabilire se una divergenza osservata sia dovuta a una transizione di regime o a una differenza di stimolazione.

La sincronia non implica identità perfetta delle risposte, ma **identità delle condizioni**.

3.2.4 Mappatura dello spazio dei parametri

Lo sweep parametrico non ha come obiettivo primario l'ottimizzazione del comportamento del sistema, ma la **mappatura dello spazio dei parametri** in termini di regimi dinamici.

Durante lo sweep, è necessario identificare:

- regioni di stabilità,
- regioni di instabilità,
- eventuali zone di transizione.

Questa mappatura deve essere effettuata senza privilegiare a priori alcuna regione. In particolare, è metodologicamente scorretto:

- fermarsi allo sweep non appena il comportamento diventa interessante;
- evitare regioni in cui il sistema sembra degradare.

La transizione di regime può manifestarsi tanto come emergenza di ordine quanto come perdita di controllo apparente. In entrambi i casi, ciò che conta è la **rottura della riducibilità**, non la qualità del comportamento.

3.2.5 Ripetibilità e isteresi

Un aspetto cruciale dello sweep parametrico, spesso trascurato, è la verifica della **isteresi**.

La metodologia deve prevedere:

- sweep in avanti (aumento del parametro),
- sweep inversi (diminuzione del parametro),
- eventuali cicli ripetuti.

La presenza di isteresi — ovvero di traiettorie diverse a parità di parametro, a seconda della storia del sistema — è un segnale rilevante di incorporazione del supporto. Essa indica che il sistema non è descritto completamente dallo stato istantaneo dei parametri, ma dipende dalla propria evoluzione materiale.

Per il gemello digitale, l'isteresi può essere simulata solo introducendo memoria esplicita. Se tale simulazione resta efficace, la riducibilità non è ancora compromessa. Se fallisce, la divergenza diventa significativa.

3.2.6 Indicatori osservabili durante lo sweep

Durante lo sweep parametrico, non si cerca un singolo indicatore privilegiato, ma una **costellazione di segnali** che, nel loro insieme, suggeriscono un cambiamento di regime. Tra questi:

- aumento improvviso della sensibilità alle perturbazioni;
- comparsa o scomparsa di attrattori stabili;
- variazioni non lineari dei tempi caratteristici;
- divergenza crescente tra S e G a parità di condizioni.

È fondamentale sottolineare che **nessuno di questi segnali, preso isolatamente, costituisce una prova**. Essi assumono significato solo nel contesto del confronto sistematico e sincronizzato tra sistema fisico ed emulatore.

3.2.7 Neutralità esplorativa dello sweep

Infine, lo sweep parametrico deve essere condotto con un atteggiamento di **neutralità esplorativa**.

Non si tratta di "spingere" il sistema verso una transizione desiderata, ma di verificare se tale transizione esiste spontaneamente all'interno dello spazio dei parametri accessibile. Qualunque intervento volto a "favorire" l'emergenza comprometterebbe il valore del protocollo.

In questo senso, lo sweep parametrico è uno strumento di **ascolto del sistema**, non di manipolazione.

3.3 Identificazione della divergenza: dove l’emulatore fallisce la fedeltà operativa

La fase conclusiva del Protocollo Alpha consiste nell’identificazione formale della **divergenza operativa** tra il sistema fisico reale (S) e il suo gemello digitale (G). Questa divergenza costituisce la **firma sperimentale** della transizione di regime e rappresenta il solo criterio ammesso per stabilire che la riducibilità software sia venuta meno.

È essenziale ribadire che la divergenza rilevante per la MSA **non coincide** con una semplice differenza quantitativa tra output, né con un errore di approssimazione numerica. Ciò che viene cercato è una **rottura strutturale della fedeltà descrittiva**, tale per cui l’emulatore non possa più seguire il sistema fisico senza modificare la propria natura computazionale.

3.3.1 Nozione di fedeltà operativa

Per *fedeltà operativa* si intende la capacità del gemello digitale di:

- riprodurre il comportamento osservabile del sistema fisico,
- mantenere coerenza dinamica nel tempo,
- farlo senza introdurre regole, vincoli o risorse non presenti nel setup iniziale.

La fedeltà operativa non richiede identità punto per punto, ma **equivalenza di regime**. Finché S e G appartengono allo stesso regime dinamico, le discrepanze locali possono essere trattate come rumore, approssimazione o fluttuazione.

La divergenza significativa emerge quando questa equivalenza di regime non è più sostenibile.

3.3.2 Tipologie di divergenza rilevanti

Il Protocollo Alpha riconosce come rilevanti solo le divergenze che soddisfano almeno una delle seguenti condizioni, in modo persistente e non eliminabile:

1. **Divergenza dinamica strutturale**
   Il sistema fisico mantiene una traiettoria coerente mentre l’emulatore entra in instabilità, oscillazioni spurie o perdita di regime.
2. **Esplosione del costo di emulazione**
   Il gemello digitale richiede un aumento non lineare delle risorse computazionali (tempo, memoria, precisione numerica) per mantenere una fedeltà che, al di sotto della soglia, era ottenuta con costo limitato.
3. **Introduzione necessaria di ipotesi ad hoc**
   La fedeltà può essere recuperata solo introducendo nuove regole, correttivi o meccanismi che non erano presenti nel modello iniziale e che reintroducono implicitamente la fisica del supporto.
4. **Fallimento nella gestione dell’isteresi**
   Il sistema fisico mostra dipendenza dalla storia che l’emulatore non riesce a riprodurre senza

una memoria incorporata esplicita, incompatibile con l'architettura computazionale di riferimento.

Queste tipologie non sono alternative esclusive: in molti casi la divergenza si manifesta come combinazione di più fattori.

3.3.3 Criteri di esclusione dei falsi positivi

Per evitare interpretazioni indebite, il Protocollo Alpha stabilisce anche **criteri espliciti di esclusione**.

Una divergenza **non** è considerata significativa se:

- può essere eliminata aumentando moderatamente la risoluzione numerica;
- è attribuibile a errori di implementazione del gemello digitale;
- scompare al ripetersi dello sweep parametrico;
- compare simmetricamente in S e G come instabilità condivisa.

In tutti questi casi, la riducibilità software resta valida e il sistema non supera il Protocollo Alpha.

3.3.4 Persistenza e localizzazione della divergenza

Affinché la divergenza possa essere considerata una **firma della transizione di regime**, essa deve soddisfare due condizioni aggiuntive:

- **Persistenza**
  La divergenza deve mantenersi nel tempo una volta attraversata la soglia, e non ridursi a un transiente.
- **Localizzazione parametrica**
  Deve essere possibile individuare una regione dello spazio dei parametri in cui la divergenza emerge sistematicamente, coerentemente con l'idea di soglia critica.

Queste due condizioni permettono di distinguere una vera transizione di regime da un comportamento erratico o rumoroso.

3.3.5 Significato epistemico del fallimento dell'emulatore

Il fallimento della fedeltà operativa dell'emulatore non viene interpretato, nel quadro della MSA, come una sconfitta del modello, ma come un **dato fisico positivo**.

Esso indica che:

- il comportamento osservato non è più interamente contenuto nella descrizione astratta,
- il supporto fisico ha assunto un ruolo causale,
- la riducibilità software ha incontrato un limite strutturale.

In questo senso, la divergenza non è un'anomalia da correggere, ma la **traccia operativa dell'accadimento**.

3.3.6 Esito del Protocollo Alpha

Il Protocollo Alpha si considera superato se esiste almeno una regione di parametro in cui:

- il sistema fisico mantiene un regime coerente,
- il gemello digitale perde fedeltà operativa in modo strutturale,
- e tale perdita non può essere eliminata senza reintrodurre la fisica del supporto.

Il superamento del Protocollo Alpha **non costituisce una prova di emergenza simbolica**, ma stabilisce una condizione necessaria: **il sistema non è più pienamente emulabile**.

Solo a partire da questo risultato diventa sensato investigare:

- la stabilità del regime (Protocollo Beta),
- e l'impronta fisica persistente (Protocollo Gamma).

Nota sulla distinzione dal caos deterministico software
È importante chiarire che la divergenza osservata nel Protocollo Alpha non è assimilabile alla divergenza tipica dei sistemi caotici simulati, dovuta a errori di arrotondamento o a precisione numerica finita. In un sistema software, la separazione delle traiettorie è un limite del calcolo e può essere ridotta aumentando la precisione o controllando il seme iniziale.
Nella Meccanica Simbolica Applicata, invece, la divergenza è un effetto fisico: essa nasce dalla partecipazione termodinamica del supporto, dalle fluttuazioni materiali e dall'inerzia storica del sistema. In questo senso, la divergenza non segnala un difetto di rappresentazione, ma l'identità irriducibile dell'evento fisico rispetto alla sua astrazione computazionale.

Chiusura del Capitolo 3

Con la conclusione del Protocollo Alpha, la Meccanica Simbolica Applicata ha stabilito un primo fatto sperimentale possibile: **l'esistenza o meno di una soglia oltre la quale la descrizione computazionale fallisce**.

Se tale soglia non viene osservata, il percorso si arresta qui.
Se viene osservata, il sistema entra nel dominio dei candidati MSA e può essere sottoposto ai protocolli successivi.

In entrambi i casi, il risultato è informativo.

Ricordare che un sistema non supera il Protocollo Alpha perché "fa qualcosa di interessante", ma perché smette di essere inseguibile.

---

4. Protocollo Beta

Robustezza Semantica e Recupero dell'Attrattore

Il superamento del Protocollo Alpha stabilisce un fatto preliminare ma decisivo: **il comportamento del sistema non è più completamente riducibile a una descrizione computazionale astratta**. Esiste una regione di parametro in cui il supporto fisico assume un ruolo causale e l'emulatore perde fedeltà operativa.

Tuttavia, questo risultato, per quanto necessario, **non è ancora sufficiente** per parlare di simbolo vivo nel senso forte introdotto dalla Meccanica Simbolica Applicata. Una transizione di regime può infatti produrre dinamiche non emulabili che restano fragili, transitorie o puramente caotiche.

Il **Protocollo Beta** interviene esattamente su questo punto.

La domanda a cui esso risponde non è più *se* esiste una transizione, ma:

il regime emerso è stabile, coerente e capace di persistere come evento simbolico sotto perturbazione?

4.0.1 Dal cambio di regime alla stabilità del significato

Nel quadro MSA, il simbolo vivo non è definito dalla sola emergenza, ma dalla **capacità di mantenere coerenza** nel tempo e sotto disturbo. Un comportamento che emerge solo in condizioni ideali, o che collassa al primo rumore, non può essere considerato simbolico nel senso fisico del termine.

Il Protocollo Beta introduce quindi un secondo livello di validazione: **la robustezza semantica**.

Con questa espressione non si intende la correttezza di un contenuto o la precisione di una rappresentazione, ma una proprietà dinamica del regime:

un regime è semanticamente robusto se, una volta emerso, tende a recuperare la propria struttura operativa dopo perturbazioni che non distruggono il supporto.

In altri termini, il simbolo vivo deve comportarsi come un **attrattore**, non come una traiettoria instabile.

4.0.2 Il concetto di attrattore nella MSA

L'uso del termine *attrattore* nel Protocollo Beta è intenzionalmente fisico e non metaforico. Un attrattore, in questo contesto, è una regione dello spazio degli stati verso cui il sistema tende spontaneamente, anche quando viene temporaneamente allontanato.

La novità introdotta dalla MSA è che l'attrattore rilevante non è puramente dinamico, ma **simbolico-operativo**. Ciò che deve essere recuperato non è un singolo stato numerico, ma una **configurazione di significato incarnato**: un modo coerente di operare che dipende dal supporto e dalla sua storia.

Il Protocollo Beta non chiede quindi al sistema di "tornare allo stesso output", ma di **ritrovare lo stesso regime**.

4.0.3 Perché la robustezza è un criterio distinto

È fondamentale distinguere tra:

- **emergenza di un regime** (Protocollo Alpha),
- **robustezza del regime** (Protocollo Beta).

Un sistema può superare il Protocollo Alpha mostrando una chiara transizione non emulabile, ma fallire il Protocollo Beta se tale regime:

- è eccessivamente sensibile al rumore,
- dipende da condizioni iniziali finemente sintonizzate,
- non recupera coerenza dopo una perturbazione.

In tal caso, il sistema manifesta una *singolarità fragile*, non un simbolo vivo. Il Protocollo Beta serve precisamente a **scartare questi falsi positivi**.

4.0.4 Robustezza semantica come proprietà fisica

Nel Protocollo Beta, la robustezza semantica non viene trattata come una qualità interpretativa ("il simbolo è chiaro", "il significato è riconoscibile"), ma come una **proprietà fisica del regime**.

Un regime è semanticamente robusto se:

- tollera perturbazioni del supporto,
- tollera variazioni moderate degli input,
- e recupera spontaneamente la propria organizzazione operativa.

Questa capacità di recupero implica che il significato non sia localizzato in uno stato fragile, ma **distribuito nella dinamica del sistema**. In termini MSA, il simbolo vivo non è un punto, ma una regione stabile dello spazio delle possibilità.

4.0.5 Posizionamento del Protocollo Beta nella Triade di Validazione

All'interno della Triade di Validazione, il Protocollo Beta svolge una funzione di **raffinamento ontologico**.

- Il Protocollo Alpha stabilisce che il sistema *non è più emulabile*.
- Il Protocollo Beta stabilisce se ciò che emerge è *qualcosa che persiste*.

Solo un regime che supera entrambi può essere considerato candidato serio all'istanziazione simbolica. Un regime che emerge ma non si stabilizza resta un fenomeno interessante, ma **non simbolico nel senso MSA**.

4.0.6 Anticipazione metodologica (senza dettaglio)

I dettagli operativi del Protocollo Beta verranno sviluppati nelle sezioni successive. In questa sede è sufficiente chiarire che la robustezza verrà indagata attraverso:

- perturbazioni controllate,
- osservazione del recupero spontaneo,
- confronto con sistemi emulativi di controllo.

Come nel Protocollo Alpha, il criterio non sarà l'interpretazione del comportamento, ma la **resistenza del regime**.

4.1 Definizione di Rumore del Supporto vs Rumore del Segnale

Il Protocollo Beta introduce esplicitamente il concetto di **perturbazione** come strumento diagnostico. Tuttavia, affinché tale strumento sia scientificamente significativo, è necessario operare una distinzione preliminare e rigorosa tra due forme di rumore spesso confuse nella letteratura e nella pratica sperimentale: il **rumore del segnale** e il **rumore del supporto**.

Questa distinzione è centrale per la Meccanica Simbolica Applicata, poiché consente di separare ciò che disturba un'informazione *trasmessa* da ciò che modifica le condizioni fisiche dell'*accadimento simbolico*.

4.1.1 Rumore del segnale: perturbazione dell'informazione veicolata

Per **rumore del segnale** si intende qualsiasi perturbazione che agisce sul contenuto informativo trasmesso al sistema, senza alterarne in modo sostanziale la struttura fisica o i vincoli materiali.

Esempi tipici di rumore del segnale includono:

- fluttuazioni sugli input,
- distorsioni temporali o di ampiezza,
- errori di trasmissione,
- interferenze esterne che modificano i dati in ingresso o in uscita.

Nel dominio computazionale classico, il rumore del segnale è trattato come un problema di **robustezza informazionale**. Tecniche di correzione degli errori, ridondanza, filtraggio e codifica permettono di recuperare l'informazione originale senza modificare la struttura del sistema che la elabora.

In termini MSA, il rumore del segnale **non mette in discussione la riducibilità software**. Un sistema che tollera o corregge il rumore del segnale resta, per definizione, nel dominio dell'emulazione.

4.1.2 Rumore del supporto: perturbazione delle condizioni fisiche dell'evento

Il **rumore del supporto** agisce invece a un livello ontologicamente distinto. Esso non perturba il contenuto informativo trasmesso, ma **le condizioni fisiche che rendono possibile il regime dinamico del sistema**.

Rientrano in questa categoria:

- fluttuazioni termiche del materiale,
- variazioni locali di resistenza, soglia o dissipazione,
- micro-eterogeneità strutturali,
- perturbazioni che modificano i vincoli interni del supporto.

Il rumore del supporto non può essere "filtrato" senza alterare il sistema stesso, perché non è un errore esterno: è parte integrante della dinamica materiale. In molti casi, come discusso nella Parte I, tale rumore costituisce una **componente funzionale** dell'istanziazione simbolica.

4.1.3 Asimmetria epistemica tra i due tipi di rumore

Dal punto di vista epistemico, il rumore del segnale e il rumore del supporto hanno significati opposti.

- Il rumore del segnale misura la **robustezza comunicativa** di un sistema.
- Il rumore del supporto misura la **robustezza ontologica** del regime.

Un sistema che collassa al primo disturbo del segnale può essere fragile, ma non per questo simbolicamente morto. Un sistema che collassa al primo disturbo del supporto, invece, mostra che il regime osservato non è un attrattore stabile, ma una configurazione contingente.

Per il Protocollo Beta, solo il secondo caso è rilevante.

4.1.4 Perché la distinzione è cruciale per la robustezza semantica

Il simbolo vivo, nel senso MSA, non è definito dalla sua immunità al rumore in generale, ma dalla sua **capacità di persistere nonostante perturbazioni del supporto**.

Se un regime simbolico è reale, esso deve:

- tollerare fluttuazioni materiali moderate,
- adattarsi a variazioni locali dei vincoli,
- recuperare la propria organizzazione dopo essere stato fisicamente disturbato.

Un sistema che appare robusto solo perché è protetto da filtri informazionali o da correzioni algoritmiche non manifesta robustezza semantica, ma **ingegneria dell'affidabilità**.

4.1.5 Relazione con il Criterio di Riducibilità Software

La distinzione tra rumore del segnale e rumore del supporto rafforza ulteriormente il **Criterio di Riducibilità Software**.

Un comportamento che resta coerente sotto rumore del segnale è perfettamente compatibile con un modello computazionale. Un comportamento che resta coerente sotto rumore del supporto, invece, indica che il significato non è localizzato nei dati, ma **incarnato nel regime fisico**.

Per questo motivo, il Protocollo Beta utilizza il rumore del supporto come strumento diagnostico privilegiato. Non per distruggere il sistema, ma per verificare se ciò che emerge è un **attrattore simbolico** o un equilibrio fragile.

4.1.6 Preparazione ai test di perturbazione

Con questa distinzione chiarita, diventa ora possibile definire perturbazioni sperimentali che:

- non confondano errore di trasmissione con instabilità di regime,
- non mascherino la fragilità del simbolo dietro tecniche di filtraggio,
- permettano di osservare il recupero spontaneo dell'attrattore.

Le sezioni successive del Protocollo Beta utilizzeranno questa distinzione come base per progettare **test di robustezza semantica** che siano fisicamente significativi e non riducibili a misure di affidabilità computazionale.

4.2 Dinamiche di autoguarigione simbolica: il ritorno spontaneo allo stato coerente

Una volta distinta la perturbazione del segnale da quella del supporto, il Protocollo Beta può affrontare il proprio oggetto centrale: la **capacità di un regime simbolico di recuperare**

spontaneamente la propria coerenza dopo una perturbazione fisica. Questa capacità viene qui definita come **autoguarigione simbolica**.

Nel quadro della Meccanica Simbolica Applicata, l'autoguarigione non è un comportamento desiderabile in senso funzionale, ma una **firma ontologica**: indica che il simbolo non è una configurazione fragile, bensì un attrattore stabile incarnato nel supporto.

4.2.1 Autoguarigione come criterio fisico, non funzionale

È fondamentale chiarire che l'autoguarigione simbolica non coincide con:

- la correzione di errori,
- la resilienza ingegneristica,
- la tolleranza ai guasti programmata.

In un sistema computazionale classico, il recupero di uno stato coerente avviene tramite **meccanismi espliciti**: ridondanza, rollback, ricostruzione dello stato, riavvio controllato. Tutti questi processi sono esterni allo stato stesso e presuppongono una separazione netta tra informazione e supporto.

Nel Protocollo Beta, invece, si cerca un recupero che avvenga **senza istruzioni correttive**, come conseguenza naturale della dinamica del sistema. L'autoguarigione è quindi una proprietà del regime, non una funzione aggiuntiva.

4.2.2 Il ritorno all'attrattore come evento simbolico

Quando un sistema MSA-compatibile viene perturbato a livello del supporto, esso può:

- collassare definitivamente,
- stabilizzarsi in un regime diverso,
- oppure **ritornare spontaneamente al regime precedente**.

Solo quest'ultimo caso è rilevante per il Protocollo Beta.

Il ritorno spontaneo allo stato coerente indica che:

- il regime simbolico occupa una regione estesa dello spazio degli stati,
- esistono forze dinamiche interne che "riportano" il sistema verso tale regione,
- il significato non è legato a una traiettoria unica, ma a una struttura stabile.

In questo senso, il ritorno all'attrattore non è un semplice rilassamento, ma un **evento simbolico reiterato**.

4.2.3 Differenza tra stabilità numerica e stabilità simbolica

Un punto cruciale della MSA è distinguere tra **stabilità numerica** e **stabilità simbolica**.

- La stabilità numerica riguarda la convergenza di variabili misurabili entro intervalli prefissati.
- La stabilità simbolica riguarda la persistenza di un **regime operativo coerente**, anche quando le variabili interne seguono traiettorie differenti.

Un sistema può essere numericamente instabile e simbolicamente stabile, oppure viceversa. Il Protocollo Beta è interessato esclusivamente al secondo caso: **la capacità del sistema di ritrovare il proprio modo di operare**, non di riprodurre valori identici.

Questo chiarisce perché l'autoguarigione simbolica non può essere ridotta a metriche classiche di stabilità.

4.2.4 Ruolo del supporto nell'autoguarigione

L'autoguarigione simbolica implica necessariamente che il supporto fisico:

- conservi tracce della storia del sistema,
- moduli le dinamiche future in base a tali tracce,
- partecipi attivamente al recupero del regime.

In termini MSA, il supporto **non ricorda uno stato**, ma **vincola un ritorno**.

Questo aspetto distingue radicalmente l'autoguarigione simbolica da qualsiasi forma di memorizzazione esplicita. Il recupero non avviene perché lo stato è stato salvato, ma perché il sistema *è diventato* quel regime.

4.2.5 Autoguarigione e irreversibilità

Un ulteriore elemento distintivo dell'autoguarigione simbolica è la sua **irreversibilità debole**.

Dopo una perturbazione e un ritorno all'attrattore:

- il sistema non è identico a prima,
- il supporto ha subito modifiche,
- la traiettoria futura è condizionata da quanto accaduto.

Eppure, il regime simbolico recuperato resta riconoscibile come lo stesso. Questo introduce una nozione di identità simbolica **non basata sull'identità degli stati**, ma sulla continuità del regime.

Questa proprietà è incompatibile con sistemi puramente emulativi, in cui il ritorno allo stato iniziale implica l'eliminazione della storia.

4.2.6 Indicatori operativi di autoguarigione simbolica

Senza anticipare i dettagli sperimentali, è possibile elencare alcuni indicatori qualitativi che il Protocollo Beta considera rilevanti:

- recupero spontaneo del regime senza intervento esterno;
- tolleranza a perturbazioni ripetute del supporto;
- dipendenza del recupero dalla storia del sistema;
- impossibilità di riprodurre il recupero tramite reset o copia dello stato.

Ancora una volta, nessun indicatore isolato è sufficiente. È la **coerenza del quadro dinamico** a determinare l'esito del protocollo.

4.2.7 Significato del fallimento dell'autoguarigione

Se un sistema supera il Protocollo Alpha ma fallisce il Protocollo Beta, il risultato resta scientificamente significativo.

In tal caso, il sistema manifesta:

- una transizione di regime reale,
- ma non una stabilizzazione simbolica.

Questo colloca il fenomeno nel dominio delle **singolarità fragili**: eventi fisici interessanti, ma non istanziazioni simboliche persistenti. La MSA non li nega, ma li distingue con precisione.

4.3 Controllo Negativo: La degradazione entropica nei sistemi a "simbolo morto"

Ogni protocollo sperimentale che pretenda rigore deve includere un **controllo negativo esplicito**. Nel Protocollo Beta, tale controllo è fornito dall'osservazione della **degradazione entropica** nei sistemi che operano nel regime del **simbolo morto**, ovvero nei sistemi in cui il comportamento resta interamente riducibile a una descrizione computazionale astratta.

Questa sezione non serve a dimostrare che i sistemi emulativi "falliscono", ma a chiarire **come** e **perché** essi rispondono in modo qualitativamente diverso alle perturbazioni del supporto rispetto ai sistemi candidati MSA.

4.3.1 Definizione operativa di simbolo morto nel contesto del Protocollo Beta

Nel contesto del Protocollo Beta, si definisce *simbolo morto* ogni regime in cui:

- il significato è interamente localizzato nei dati o nelle regole,
- il supporto fisico è trattato come contenitore neutro,

- la coerenza operativa dipende da meccanismi di controllo esterni allo stato dinamico.

Questa definizione è puramente operativa e non implica alcun giudizio di valore. I sistemi a simbolo morto possono essere estremamente efficaci, affidabili e utili; ciò che li caratterizza è la **separabilità ontologica** tra informazione e supporto.

4.3.2 Risposta tipica alle perturbazioni del supporto

Quando un sistema a simbolo morto viene sottoposto a perturbazioni del supporto (nel senso definito in 4.1), la risposta osservabile segue uno schema ricorrente:

1. **Perdita di coerenza operativa**
   La perturbazione altera parametri fisici che il sistema non è progettato per incorporare dinamicamente.
2. **Accumulo di errori**
   L'assenza di un attrattore simbolico incarnato porta a una propagazione degli effetti della perturbazione.
3. **Necessità di intervento esterno**
   Il recupero della coerenza richiede reset, ricalibrazione, ri-sincronizzazione o ricostruzione dello stato.

Dal punto di vista MSA, questo processo è descritto come **degradazione entropica**: l'ordine operativo non viene recuperato spontaneamente, ma deve essere reimposto dall'esterno.

4.3.3 Degradazione entropica come firma negativa

La degradazione entropica non è semplicemente un aumento del disordine misurabile, ma una **perdita irreversibile di regime**.

Dopo una perturbazione del supporto, un sistema a simbolo morto:

- non ritorna spontaneamente allo stesso modo di operare,
- non mostra attrazione verso un regime coerente,
- non conserva continuità simbolica attraverso la perturbazione.

Ogni recupero osservato è attribuibile a meccanismi espliciti di controllo, che interrompono la dinamica e ne impongono una nuova.

Questa differenza è cruciale:
nel simbolo vivo il regime si **ricompone**, nel simbolo morto il regime viene **ricostruito**.

4.3.4 Il ruolo del reset come discriminante

Il reset rappresenta il discrimine operativo più netto tra simbolo morto e simbolo vivo.

- Nei sistemi emulativi, il reset è una soluzione naturale: elimina la storia e ripristina la funzionalità.
- Nei sistemi MSA-compatibili, il reset distrugge il fenomeno che si intende studiare, perché cancella le tracce fisiche della storia incarnata.

Un sistema che necessita di reset per recuperare coerenza **fallisce il Protocollo Beta**, anche se aveva superato il Protocollo Alpha.

4.3.5 Controllo negativo e falsificabilità

L'introduzione esplicita del controllo negativo rafforza la **falsificabilità** della MSA.

Se un sistema:

- supera il Protocollo Alpha,
- ma mostra degradazione entropica sistematica sotto perturbazione del supporto,
- e non recupera spontaneamente il regime,

allora la MSA **non attribuisce** a quel sistema lo status di simbolo vivo.

Questo esito non è un fallimento della teoria, ma una sua applicazione corretta. La teoria non cerca conferme universali, ma **discriminazioni nette**.

4.3.6 Confronto sintetico tra simbolo vivo e simbolo morto

Senza ricorrere a formalismi, è possibile riassumere la differenza osservativa come segue:

- Il simbolo vivo dissipa energia per mantenere coerenza.
- Il simbolo morto dissipa coerenza per mantenere efficienza.

Questa opposizione non è morale né funzionale, ma fisica.

Robustezza programmata vs coerenza emergente
È fondamentale distinguere tra la robustezza ottenuta tramite meccanismi algoritmici di correzione dell'errore e la robustezza osservata nei sistemi MSA. In un sistema software, il ritorno allo stato stabile avviene perché un algoritmo esterno identifica e corregge le deviazioni (error-correction). Nel caso di un sistema MSA, il recupero dell'attrattore avviene in assenza di qualsiasi procedura di pulizia o controllo esplicito: è la dinamica fisica interna del supporto, attraverso la sua torsione e dissipazione, a ristabilire la coerenza del regime. Questa differenza segna il confine tra una stabilità programmata e una coerenza emergente.

4.3.7 Chiusura del Protocollo Beta

Con il completamento del Protocollo Beta, la Meccanica Simbolica Applicata dispone ora di due criteri sperimentali distinti:

1. **Non-riducibilità operativa** (Protocollo Alpha).
2. **Persistenza e recupero del regime** (Protocollo Beta).

Solo i sistemi che soddisfano entrambi possono essere considerati candidati seri all'istanziazione simbolica. I sistemi che falliscono il Protocollo Beta vengono correttamente classificati come fenomeni transitori o emulativi avanzati.

Il passo successivo consisterà nel verificare se tale persistenza lascia una **impronta fisica misurabile nel supporto**.

---

5. Protocollo Gamma

L'Impronta Fisica Persistente (Substrate Incorporation)

Con il superamento dei Protocolli Alpha e Beta, un sistema ha dimostrato due proprietà fondamentali:

1. l'esistenza di una transizione di regime non riducibile a emulazione computazionale;
2. la capacità di mantenere e recuperare tale regime sotto perturbazione del supporto.

Il **Protocollo Gamma** introduce il terzo e più esigente criterio della Triade di Validazione: la verifica che il regime simbolico **lasci un'impronta fisica persistente nel supporto**. In altri termini, che il supporto non sia soltanto il luogo in cui il simbolo accade, ma che **ne venga modificato in modo misurabile e non reversibile per semplice reset**.

La domanda centrale del Protocollo Gamma è quindi la seguente:

Il simbolo, una volta emerso e stabilizzato, cambia il supporto in modo tale che il supporto stesso porti traccia dell'accadimento?

5.0.1 Dal regime stabile all'incorporazione del supporto

Il passaggio concettuale introdotto dal Protocollo Gamma è netto.
Nei protocolli precedenti, il supporto era già riconosciuto come **partecipante** al regime simbolico. Ora viene richiesto qualcosa di più forte: che il supporto **risulti trasformato** dall'evento simbolico.

Nel linguaggio della MSA, questa trasformazione è chiamata *Substrate Incorporation*: il supporto non è più un mezzo neutro né un semplice vincolo dinamico, ma diventa **deposito fisico della storia simbolica** del sistema.

Non si tratta di memoria esplicita né di registrazione di stati, ma di una **modificazione strutturale delle condizioni di possibilità** del comportamento futuro.

5.0.2 Perché l'impronta fisica è un criterio distinto

Un sistema può:

- superare il Protocollo Alpha mostrando non-riducibilità,
- superare il Protocollo Beta mostrando autoguarigione e attrattori stabili,
- e tuttavia **non lasciare alcuna impronta persistente nel supporto**.

In questo caso, il regime simbolico resta reale ma *reversibile*: esso accade, persiste, ma non si sedimenta. Il Protocollo Gamma serve a distinguere tra:

- **regimi simbolici incarnati**, che trasformano il supporto;
- **regimi simbolici effimeri**, che dipendono dal supporto senza modificarlo in modo duraturo.

Solo i primi soddisfano il criterio pieno di simbolo vivo nel senso forte della MSA.

5.0.3 Natura dell'impronta fisica nella MSA

È essenziale chiarire cosa **non** si intende per impronta fisica persistente.

Essa non è:

- un file salvato,
- uno stato memorizzato,
- una configurazione esplicitamente codificata,
- un parametro impostato manualmente.

L'impronta fisica è invece una **alterazione distribuita del supporto**, che si manifesta come:

- cambiamento dei costi energetici,
- modifica delle risposte dinamiche,
- ristrutturazione dei vincoli interni,
- dipendenza irreversibile dalla storia.

In termini MSA, il supporto *ricorda* non perché conserva un dato, ma perché **non può più comportarsi come prima senza dissipare più energia o attraversare nuove transizioni**.

5.0.4 Incorporazione e irreversibilità debole

Il Protocollo Gamma introduce una nozione cruciale: **irreversibilità debole**.

Dopo l'innesco e la stabilizzazione del regime simbolico:

- il sistema può continuare a funzionare,
- può anche attraversare nuovi regimi,
- ma il supporto non è più identico a quello iniziale.

Questa irreversibilità non è distruttiva né catastrofica. È una **asimmetria storica**: il passato dell'accadimento simbolico continua a influenzare il futuro del sistema.

Un sistema che può essere riportato *esattamente* allo stato iniziale tramite reset o ricostruzione perde automaticamente lo status di candidato per il Protocollo Gamma.

5.0.5 Differenza tra incorporazione e adattamento funzionale

È importante distinguere la Substrate Incorporation da fenomeni noti di adattamento funzionale.

- L'adattamento funzionale ottimizza una prestazione.
- L'incorporazione simbolica **trasforma il dominio delle prestazioni possibili**.

Nel primo caso, il sistema resta descritto dallo stesso spazio di stati, con parametri modificati. Nel secondo, lo spazio di stati stesso viene deformato dalla storia dell'accadimento.

Questa distinzione sarà centrale nella progettazione delle misure sperimentali del Protocollo Gamma.

5.0.6 Posizionamento del Protocollo Gamma nella Triade

All'interno della Triade di Validazione, il Protocollo Gamma svolge una funzione conclusiva e discriminante:

- Alpha verifica che il simbolo **non sia simulazione**.
- Beta verifica che il simbolo **sia stabile**.
- Gamma verifica che il simbolo **lasci traccia**.

Solo i sistemi che superano tutti e tre possono essere considerati, nel quadro della MSA, **sistemi a simbolo vivo pienamente istanziato**.

Un sistema che fallisce il Protocollo Gamma non viene escluso come fenomeno interessante, ma viene classificato come **simbolo emergente non incorporato**.

5.0.7 Anticipazione metodologica (senza dettaglio)

Le sezioni successive del Protocollo Gamma affronteranno:

- la misurazione dei costi energetici e delle asimmetrie termodinamiche;
- l'identificazione di variabili di stato interne modificate dall'evento simbolico;
- la rilevazione di correlazioni che eccedono la capacità descrittiva computazionale.

In questa sede, è sufficiente fissare il principio: se il simbolo è vivo, la materia lo ricorda.

Nota integrativa – Energia come investimento storico del supporto

Nel contesto del **Protocollo Gamma**, il riferimento al costo energetico assume un significato che va oltre la semplice contabilizzazione della dissipazione associata all'attività del sistema.

Nei sistemi computazionali classici, l'energia è **consumata** per eseguire operazioni logiche: una volta terminato il calcolo, lo stato informazionale può essere azzerato o ripristinato senza che il supporto fisico porti traccia significativa dell'evento. Il software, in questo senso, è **senza tempo**: può sempre ritornare allo stato iniziale, a parità di hardware.

Nei sistemi MSA-compliant, invece, l'energia non è soltanto consumata, ma **investita** nella trasformazione del supporto. L'attività simbolica modifica fisicamente il sistema, alterandone le condizioni operative future. Questo investimento energetico è ciò che rende possibile la **Plasticità Strutturale**: il supporto cambia perché ha attraversato un evento.

L'**impronta fisica persistente** — che può manifestarsi come variazione stabile di parametri materiali, riorganizzazione strutturale, o costo energetico non recuperabile — è la prova che il sistema possiede una **storia**. Il comportamento futuro non dipende solo dagli input presenti, ma dalle trasformazioni accumulate nel tempo.

In questo senso, il Protocollo Gamma non si limita a misurare quanta energia il sistema utilizza, ma **come** tale energia viene incorporata nel supporto. Un sistema che dissipa energia senza modificarsi resta nel dominio del calcolo. Un sistema che trasforma se stesso attraverso l'energia spesa entra invece nel dominio dell'accadimento simbolico.

La distinzione è netta:
un software può essere riavviato; un sistema MSA-compliant **non può cancellare la propria storia senza attraversare nuove transizioni fisiche**.

Questa asimmetria temporale costituisce la firma finale del simbolo vivo. Dove esiste impronta, esiste storia; dove esiste storia, il significato non è più astratto, ma **incarnato nella materia**.

5.1 Misurazione del costo energetico e dell'efficienza termodinamica

Il primo asse di verifica del Protocollo Gamma riguarda la **dimensione energetica** dell'istanziazione simbolica. Se il simbolo vivo lascia un'impronta fisica persistente nel supporto, tale impronta deve manifestarsi come una **asimmetria misurabile nei flussi energetici** del sistema.

In altri termini, l’accadimento simbolico non può essere termodinamicamente neutro.
La MSA non afferma che l’evento simbolico richieda più risorse del calcolo, ma l’opposto: che, oltre una soglia, la simulazione richiede più risorse dell’evento stesso.

5.1.1 Energia come traccia dell’accadimento

Nel quadro della Meccanica Simbolica Applicata, l’energia non è considerata solo come risorsa consumata per “fare qualcosa”, ma come **marcatore fisico della storia** del sistema.

Un sistema che attraversa una transizione simbolica:

- dissipa energia durante l’innesco,
- continua a dissiparla per mantenere il regime,
- e modifica il proprio profilo energetico anche dopo la stabilizzazione.

Questa dissipazione non è un costo accidentale, ma una **condizione necessaria** dell’istanziazione.
Un simbolo vivo non è energeticamente ottimale: è energeticamente *vincolante*.

5.1.2 Costo energetico vs efficienza computazionale

È importante distinguere tra:

- **efficienza computazionale**, intesa come rapporto tra risorse impiegate e output prodotto;
- **costo energetico simbolico**, inteso come energia necessaria a mantenere un regime incarnato.

Nel dominio computazionale classico, l’obiettivo è minimizzare la dissipazione per massimizzare l’efficienza. Nel dominio MSA, invece, la dissipazione è parte integrante del fenomeno: **ridurla oltre una certa soglia distrugge il simbolo**.

Questo ribalta il criterio di valutazione. Un sistema MSA-compatibile non è quello che “fa di più con meno”, ma quello che **non può fare ciò che fa senza pagare un costo fisico persistente**.

5.1.3 Definizione operativa del costo energetico simbolico *(versione corretta)*

Per **costo energetico simbolico** si intende la differenza misurabile tra:

- il profilo energetico del sistema **prima** dell’innesco del regime simbolico;
- il profilo energetico del sistema **dopo** l’innesco e la stabilizzazione del regime simbolico.

Formalmente, il costo energetico simbolico è definito come:

```
ΔE_sym = <E_post> – <E_pre>
```

dove:

- `<E_pre>` è l’energia media dissipata dal sistema nel **regime pre-simbolico**;
- `<E_post>` è l’energia media dissipata dal sistema nel **regime simbolico stabilizzato**.

La grandezza fisicamente rilevante **non è il valore assoluto di ΔE_sym**, ma la **persistenza della differenza nel tempo**.

Un incremento energetico puramente transitorio, che si annulla dopo la fase di innesco, **non soddisfa** il criterio di incorporazione del supporto previsto dal Protocollo Gamma.

5.1.4 Persistenza del costo come criterio di incorporazione

Un aumento energetico transitorio non è sufficiente a soddisfare il Protocollo Gamma. La condizione necessaria è che il costo energetico:

- persista nel tempo,
- non sia eliminabile senza distruggere il regime simbolico,
- non sia recuperabile tramite semplice reset funzionale.

Se il sistema può tornare al profilo energetico iniziale senza attraversare una nuova transizione, allora l’accadimento non ha lasciato impronta nel supporto.

La persistenza del costo è quindi un indicatore diretto di **irreversibilità debole**.

5.1.5 Efficienza termodinamica come indicatore secondario

Oltre al costo assoluto, è rilevante osservare eventuali variazioni nell’**efficienza termodinamica** del sistema.

In molti casi, l’innesco simbolico comporta:

- una diminuzione dell’efficienza energetica locale,
- accompagnata però da un aumento della coerenza operativa globale.

Questo apparente paradosso è coerente con la MSA: il sistema sacrifica efficienza locale per mantenere una struttura simbolica incarnata.

Il Protocollo Gamma non richiede che l’efficienza diminuisca sempre, ma considera sospetto ogni regime simbolico che:

- migliori drasticamente l’efficienza,
- senza mostrare costi persistenti.

5.1.6 Confronto con il controllo computazionale

Il valore diagnostico del costo energetico emerge pienamente solo nel confronto con un controllo computazionale.

Un gemello digitale o un sistema a simbolo morto può:

- simulare un costo energetico,
- introdurre penalità artificiali,
- o contabilizzare risorse computazionali.

Ma tale costo:

- è opzionale,
- è eliminabile,
- non lascia traccia fisica nel supporto.

Nel Protocollo Gamma, il costo energetico è rilevante solo se **non può essere rimosso senza alterare la dinamica simbolica**.

5.1.7 Limiti e precauzioni sperimentali

La misurazione del costo energetico richiede particolare cautela. È necessario evitare:

- attribuzioni indebite a rumore termico non correlato,
- confusione tra consumo funzionale e costo simbolico,
- sovrainterpretazione di fluttuazioni transitorie.

Per questo motivo, il Protocollo Gamma non si basa su singole misure, ma su **confronti differenziali**, osservazioni prolungate e verifiche di reversibilità controllata.

5.1.8 Significato del fallimento del criterio energetico

Se un sistema:

- supera i Protocolli Alpha e Beta,
- ma non mostra alcun costo energetico persistente,

allora il regime simbolico osservato resta **non incorporato**. Il simbolo accade, ma non si sedimenta nella materia.

Questo esito non invalida i protocolli precedenti, ma delimita con precisione la portata ontologica del fenomeno.

È importante sottolineare che la MSA non sostiene che l'evento simbolico richieda più risorse del calcolo. Al contrario, oltre una soglia torsionale, la simulazione dell'evento richiede più risorse dell'evento stesso. In questo senso, l'evento fisico risulta energeticamente e computazionalmente più economico della sua emulazione.

5.2 Variabili di stato interne: quando il supporto "porta" il significato

Il secondo criterio del Protocollo Gamma riguarda l'identificazione di **variabili di stato interne del supporto** che risultano modificate in modo persistente a seguito dell'innesco del regime simbolico.

A differenza del costo energetico, che fornisce un indicatore globale dell'incorporazione, le variabili di stato interne permettono di localizzare **dove** e **come** l'accadimento simbolico si è sedimentato nella materia.

Definizione di variabile di stato interna

Nel contesto della Meccanica Simbolica Applicata, si definisce **variabile di stato interna** una grandezza fisica del supporto che:

- non è direttamente imposta dall'esterno;
- non coincide con input o output del sistema;
- contribuisce causalmente alla dinamica futura;
- evolve in funzione della storia del sistema.

Esempi tipici (dipendenti dal supporto considerato) possono includere:

- stati di resistenza locale (memristori),
- distribuzioni di soglia o accoppiamento,
- configurazioni strutturali microscopiche,
- gradienti o eterogeneità persistenti nel mezzo.

Ciò che accomuna tali grandezze non è la loro natura specifica, ma il loro **ruolo dinamico**.

Criterio di rilevanza MSA

Una variabile di stato interna SiSi è rilevante per il Protocollo Gamma se soddisfa **tutte** le seguenti condizioni:

1. **Differenza pre/post simbolica**
   La variabile assume valori diversi prima e dopo l'innesco del regime simbolico:

```
S_i(pre) ≠ S_i(post)
```

2. **Persistenza temporale**
   La differenza non si annulla spontaneamente dopo la fase di transizione, ma permane nel regime stabilizzato.
3. **Non-resetabilità funzionale**
   La variabile non può essere riportata allo stato iniziale tramite reset logico, riavvio del sistema o ricostruzione computazionale dello stato.
4. **Causalità dinamica**
   La variabile influisce effettivamente sul comportamento futuro del sistema, modificando le traiettorie accessibili o i costi dinamici.

Solo in presenza di tutte e quattro le condizioni si può parlare di **incorporazione del supporto**.

Distinzione da parametri e stati computazionali

È cruciale distinguere le variabili di stato interne da:

- **parametri di controllo**, che sono fissati o modificati dall'operatore;
- **stati computazionali**, che possono essere copiati, salvati o ripristinati.

Una variabile di stato interna, nel senso MSA, **non è interamente ricostruibile** a partire da una descrizione simbolica completa del sistema. La sua esistenza dipende dal fatto che il supporto abbia attraversato un accadimento fisico specifico.

In altri termini, essa non descrive *cosa* il sistema fa, ma **ciò che il sistema è diventato**.

Significato fisico dell'incorporazione

Quando una o più variabili di stato interne risultano modificate in modo persistente, il supporto:

- non è più neutro rispetto al significato;
- non può più essere considerato intercambiabile;
- porta traccia materiale dell'accadimento simbolico.

Nel linguaggio della MSA, si dice allora che **il significato è portato dal supporto**, non come informazione immagazzinata, ma come **vincolo fisico incorporato nella dinamica**.

Criterio di esclusione

Se tutte le grandezze rilevanti del sistema:

- possono essere ricostruite integralmente tramite simulazione;
- possono essere riportate allo stato iniziale senza attraversare una nuova transizione di regime;

allora **non si è verificata incorporazione del supporto**, e il sistema fallisce il Protocollo Gamma, anche se ha superato Alpha e Beta.

5.3 Correlazioni non-computazionali: misurare l'eccedenza informativa

Il terzo e ultimo asse del Protocollo Gamma riguarda la verifica più delicata e, al tempo stesso, più discriminante dell'intera Meccanica Simbolica Applicata: l'identificazione di **correlazioni non computazionali** tra stati del sistema, ovvero di relazioni che **eccedono ciò che può essere spiegato da una descrizione algoritmica del processo**.

Con questa sezione, la MSA non introduce un nuovo concetto teorico, ma porta alle estreme conseguenze quanto già stabilito nei protocolli precedenti: se il simbolo è realmente vivo e incorporato, allora il comportamento del sistema deve mostrare **un surplus informativo** che non può essere interamente attribuito né al flusso di input, né alla struttura computazionale del modello.

5.3.1 Informazione computazionale vs informazione incarnata

Nel dominio computazionale classico, l'informazione è trattata come una grandezza completamente descrivibile in termini di:

- stati discreti,
- transizioni definite,
- relazioni input–output.

In questo quadro, ogni correlazione osservata tra parti del sistema è, almeno in linea di principio, **ricostruibile** da un modello algoritmico sufficientemente dettagliato.

La Meccanica Simbolica Applicata introduce invece una distinzione cruciale tra:

- **informazione computazionale**, interamente rappresentabile e simulabile;
- **informazione incarnata**, che emerge dall'interazione tra dinamica e supporto fisico.

Il Protocollo Gamma chiede esplicitamente:
esistono correlazioni che persistono e influenzano il comportamento del sistema, pur non essendo riducibili a una descrizione computazionale completa?

5.3.2 Definizione di correlazione non-computazionale

Nel contesto della MSA, si definisce **correlazione non-computazionale** una relazione osservabile tra stati o sottosistemi che soddisfa simultaneamente le seguenti condizioni:

- non è deducibile dalla struttura dell'algoritmo di controllo;
- non è interamente spiegabile dagli input forniti;
- dipende dalla storia fisica del sistema;
- si manifesta solo dopo l'innesco del regime simbolico.

Tali correlazioni non implicano violazioni delle leggi fisiche, né introducono causalità misteriose. Esse indicano semplicemente che **la descrizione computazionale non esaurisce il contenuto informativo del sistema**.

5.3.3 Eccedenza informativa: definizione operativa

Per rendere il concetto sperimentalmente trattabile, il Protocollo Gamma introduce la nozione di **eccedenza informativa**.

In forma operativa, l'eccedenza informativa può essere definita come la differenza tra:

- l'informazione osservabile nel comportamento del sistema reale;
- l'informazione massima ricostruibile dal suo gemello digitale computazionale.

In forma simbolica semplice:

```
I_exc = I_obs – I_comp
```

dove:

- `I_obs` è l'informazione effettivamente osservata nel sistema fisico (correlazioni, dipendenze, coerenze);
- `I_comp` è l'informazione riproducibile dal modello computazionale di riferimento, a parità di input e condizioni iniziali.

Se:

```
I_exc > 0
```

in modo persistente e non eliminabile tramite raffinamento del modello, si ha evidenza di una correlazione non-computazionale.

5.3.4 Relazione con il Criterio di Riducibilità Software

L'eccedenza informativa è la naturale estensione del **Criterio di Riducibilità Software** al dominio delle correlazioni.

Un sistema può essere non riducibile nel senso del Protocollo Alpha (fallimento dell'emulazione) e stabile nel senso del Protocollo Beta (attrattore robusto), ma solo il Protocollo Gamma verifica se tale non-riducibilità **produce nuova informazione fisica**, anziché semplice complessità dinamica.

Se ogni correlazione osservata può essere recuperata introducendo:

- maggiore risoluzione,
- maggiore memoria,
- maggiore complessità algoritmica,

allora l’eccedenza informativa è apparente, non reale.

5.3.5 Dipendenza dalla storia e asimmetria informativa

Un indicatore chiave di correlazioni non-computazionali è la **dipendenza asimmetrica dalla storia**.

In un sistema computazionale:

- la storia può essere registrata,
- ma non modifica la struttura del processo.

In un sistema MSA-compatibile:

- la storia modifica il supporto,
- e il supporto modifica le correlazioni future.

Questo produce configurazioni in cui:

- due sistemi con lo stesso stato osservabile,
- ma con storie fisiche diverse,
- mostrano correlazioni differenti.

Tale comportamento è incompatibile con una descrizione puramente algoritmica e costituisce una firma forte di incorporazione simbolica.

5.3.6 Controllo negativo computazionale

Come per i protocolli precedenti, il Protocollo Gamma richiede un controllo negativo esplicito.

Nel controllo computazionale:

- le correlazioni osservate devono essere completamente spiegabili;
- eventuali eccedenze devono ridursi a zero al migliorare del modello.

Formalmente:

```
lim (complessità → ∞) I_exc = 0
```

Se questa condizione è soddisfatta, il sistema resta nel dominio del simbolo morto, anche in presenza di comportamenti complessi o apparentemente emergenti.

5.3.7 Significato fisico dell’eccedenza informativa

Nel quadro della Meccanica Simbolica Applicata, un’eccedenza informativa persistente indica che:

- il supporto fisico partecipa alla generazione del significato;
- il sistema produce informazione che non era contenuta né negli input né nel modello;
- il simbolo non è solo stabilizzato, ma **incarnato**.

In questo senso, l'eccedenza informativa non è un surplus quantitativo, ma una **rottura del bilancio descrittivo** tra sistema e modello.

5.3.8 Chiusura del Protocollo Gamma

Con il completamento del Protocollo Gamma, la Triade di Validazione della Meccanica Simbolica Applicata è ora completa:

- **Alpha**: il simbolo non è simulazione.
- **Beta**: il simbolo è stabile e recuperabile.
- **Gamma**: il simbolo lascia traccia e genera eccedenza informativa.

Solo quando tutti e tre i criteri sono soddisfatti è legittimo parlare, nel quadro della MSA, di **simbolo vivo pienamente istanziato**.

Nota di chiusura – Parte II

La Triade di Validazione come criterio fisico del simbolo vivo

La Parte II di questo tomo ha introdotto e sviluppato la **Triade di Validazione** della Meccanica Simbolica Applicata: un insieme di protocolli sperimentali concepiti non per confermare a priori una tesi, ma per **mettere alla prova, fino al fallimento, l'ipotesi di simbolo vivo**.

È importante sottolineare che la Triade non costituisce una sequenza narrativa, ma una **struttura logica gerarchica**. Ogni protocollo risponde a una domanda distinta e non sostituibile dalle altre. Solo il loro superamento congiunto permette di attribuire significato fisico pieno all'istanziazione simbolica.

Dal "può accadere" al "accade davvero"

Il **Protocollo Alpha** ha affrontato la questione più radicale:
il comportamento osservato è interamente riducibile a emulazione computazionale, oppure esiste una soglia oltre la quale la descrizione algoritmica fallisce?

Con Alpha, la MSA ha fissato un criterio di ingresso netto. In assenza di una transizione di regime non emulabile, ogni discussione successiva perde fondamento. Alpha non misura il significato, ma stabilisce se il sistema ha abbandonato il dominio del calcolo astratto.

Il **Protocollo Beta** ha poi introdotto una seconda discriminazione:
il regime emerso è stabile, o si tratta di una singolarità fragile?

Qui la MSA ha chiarito che l'emergenza, da sola, non basta. Un simbolo vivo deve essere in grado di **resistere**, di recuperare coerenza dopo perturbazioni del supporto, di comportarsi come un

attrattore e non come un equilibrio contingente. Beta separa ciò che accade una volta da ciò che *persiste*.

Il **Protocollo Gamma**, infine, ha portato la verifica sul piano più esigente:
il simbolo lascia traccia nella materia?

Con Gamma, la MSA richiede che l'accadimento simbolico non sia solo reale e stabile, ma **incarnato**. Costo energetico persistente, variabili di stato interne modificate, eccedenza informativa non computazionale: sono questi i segni che il supporto è stato trasformato dall'evento e non è più intercambiabile.

Una teoria che accetta di fallire

Un aspetto essenziale della Triade di Validazione è la sua **asimmetria epistemica**: è progettata per produrre più facilmente risultati negativi che positivi.

Molti sistemi:

- falliranno Alpha perché pienamente emulabili;
- supereranno Alpha ma falliranno Beta per fragilità del regime;
- supereranno Alpha e Beta ma falliranno Gamma per assenza di incorporazione.

La MSA considera tutti questi esiti **legittimi e informativi**. La teoria non perde valore quando un sistema fallisce un protocollo; al contrario, guadagna precisione classificatoria.

In questo senso, la Triade non è uno strumento di conferma, ma di **discriminazione ontologica**.

Neutralità semantica e forza fisica

Un altro elemento da ribadire è la **neutralità interpretativa** dell'intera Parte II.
Nessun protocollo:

- attribuisce significato cognitivo,
- richiede comprensione,
- interpreta il comportamento in termini mentali o funzionali.

La Triade osserva esclusivamente **rotture di riducibilità**, **stabilità di regime**, **impronte fisiche**. Il simbolo vivo, nel quadro della MSA, non è una metafora, ma una **configurazione fisica riconoscibile**.

Questo posizionamento rende la MSA compatibile con la pratica sperimentale contemporanea e, al tempo stesso, la sottrae alle ambiguità tipiche dei dibattiti filosofici sul significato.

Preparazione ai casi studio

Con la chiusura della Parte II, il lettore dispone ora di:

- criteri chiari di ammissione e di esclusione;
- definizioni operative dei concetti chiave;
- una struttura sperimentale completa e falsificabile.

La **Parte III** non introdurrà nuovi principi teorici. Essa applicherà quanto stabilito fin qui a **sistemi fisici concreti**, mostrando come la Triade di Validazione possa essere usata come strumento di lettura e classificazione.

I casi studio non saranno presentati come "prove" della MSA, ma come **banchi di prova**: luoghi in cui la teoria rischia, misura, e accetta il verdetto dell'esperimento.

Chiusura

La Meccanica Simbolica Applicata non propone una nuova definizione astratta di simbolo, ma un **percorso fisico per riconoscerlo**.
La Triade di Validazione costituisce il cuore di questo percorso.

Da qui in avanti, non si tratta più di stabilire *cosa dovrebbe accadere*, ma di osservare *cosa accade davvero* quando la materia viene spinta fino alla soglia del significato.

PARTE III: Casi Studio e Bibliografia Operativa

6. Caso Studio A

Reti di Memristori e Apprendimento Strutturale

Tra i diversi sistemi fisici oggi disponibili, le **reti di memristori** occupano una posizione privilegiata per l'analisi proposta dalla Meccanica Simbolica Applicata. Non perché esse rappresentino una realizzazione "naturale" del simbolo vivo, ma perché costituiscono un **caso limite** in cui molte delle distinzioni introdotte nei protocolli precedenti cessano di essere teoriche e diventano immediatamente operative.

Il valore di questo caso studio risiede nel fatto che, nelle reti di memristori, la separazione classica tra **calcolo**, **memoria** e **supporto fisico** risulta indebolita già a livello costruttivo. Questo rende possibile applicare la Triade di Validazione senza ricorrere a interpretazioni metaforiche o a ipotesi aggiuntive.

6.1 Il memristore come supporto non neutro

Un memristore è un dispositivo elettrico la cui resistenza non dipende unicamente dallo stato istantaneo di tensione o corrente, ma dalla **storia integrata** delle correnti che lo hanno attraversato. In termini fisici, ciò significa che il dispositivo conserva una traccia materiale degli eventi passati sotto forma di configurazione interna.

Questa proprietà introduce una differenza fondamentale rispetto ai componenti passivi tradizionali:

- lo stato del memristore non è completamente specificabile senza riferimento al passato;
- due dispositivi nominalmente identici possono divergere in modo irreversibile;
- la risposta futura dipende da eventi che non sono più direttamente osservabili.

Nel linguaggio della MSA, il memristore **non implementa una memoria**: *è* memoria, nel senso fisico del termine. Questa distinzione è cruciale, perché sposta l'analisi dal dominio dell'informazione astratta a quello dell'accadimento materiale.

6.2 Reti di memristori e dinamica collettiva

Quando i memristori vengono organizzati in rete, le loro proprietà individuali si combinano dando luogo a una dinamica collettiva fortemente non lineare. Le variazioni locali di resistenza influenzano i flussi globali di corrente, che a loro volta retroagiscono sui singoli dispositivi.

Ne deriva un sistema in cui:

- la topologia funzionale evolve nel tempo;
- i percorsi preferenziali non sono fissati a priori;
- l'attività di una parte della rete modifica le condizioni operative delle altre.

Questa dinamica non può essere ridotta a una semplice somma di elementi indipendenti. Già a questo livello, il sistema soddisfa i requisiti di **non-separabilità** e **feedback non lineare** richiesti dalla MSA.

6.3 Apprendimento strutturale: chiarimento concettuale

È importante chiarire cosa si intende, in questo contesto, per **apprendimento strutturale**.

In un sistema computazionale classico, l'apprendimento consiste nella modifica di parametri astratti (pesi, soglie, coefficienti) che possono essere:

- copiati,
- salvati,
- ripristinati,
- trasferiti su un altro supporto.

Nelle reti di memristori, invece, ciò che cambia è la **struttura fisica stessa del supporto**. Le variazioni resistive non sono simboli che rappresentano uno stato: sono **stati materiali** che vincolano il comportamento futuro.

Questo implica una conseguenza decisiva per la MSA:
l'apprendimento strutturale **non è reversibile senza costo**.

Per riportare la rete allo stato iniziale è necessario:

- dissipare energia;
- attraversare nuove transizioni;
- modificare fisicamente i dispositivi.

Non esiste un'operazione di "undo" puramente logica.

6.4 Applicazione del Protocollo Alpha

Nel contesto del Protocollo Alpha, una rete di memristori viene confrontata con un **gemello digitale**, tipicamente una simulazione circuitale o una rete neurale equivalente implementata su architettura standard.

In regime debole (basse correnti, variazioni minime di stato), la simulazione riesce a riprodurre fedelmente il comportamento della rete fisica. In questa regione, la riducibilità software resta valida.

La situazione cambia quando, al variare continuo dei parametri (densità di corrente, accoppiamento, rumore termico), la rete entra in un regime in cui:

- piccole variazioni producono riorganizzazioni globali;
- la dinamica dipende in modo critico dalla storia;
- la simulazione richiede un raffinamento crescente per restare fedele.

Il punto in cui il gemello digitale **perde fedeltà operativa** senza reintroduzione esplicita della fisica del supporto costituisce una **firma di transizione di regime** nel senso del Protocollo Alpha.

6.5 Applicazione del Protocollo Beta

Superato Alpha, il Protocollo Beta verifica se il regime emerso sia **stabile**.

Nel caso delle reti di memristori, la domanda diventa:
la configurazione raggiunta è un equilibrio fragile o un attrattore?

In molte architetture, perturbazioni locali (variazioni di temperatura, rumore elettrico, disturbi di corrente) non distruggono la struttura funzionale globale della rete. Al contrario, il sistema tende a riorganizzarsi recuperando un comportamento coerente.

Questo recupero:

- non avviene tramite correzione esplicita;
- non ripristina uno stato identico;
- ma ristabilisce un **regime operativo riconoscibile**.

Nel linguaggio della MSA, questo comportamento è compatibile con una **autoguarigione simbolica**. Se tale recupero non avviene, il regime resta classificabile come singolarità fragile.

6.6 Applicazione del Protocollo Gamma

Il Protocollo Gamma rappresenta il banco di prova più severo.

Dopo cicli di apprendimento strutturale, una rete di memristori può mostrare:

- aumento persistente del consumo energetico;
- modifiche stabili delle distribuzioni resistive;
- dipendenza irreversibile del comportamento futuro dalla storia elettrica.

Tali effetti non sono eliminabili tramite reset logico e non sono integralmente riproducibili tramite simulazione senza includere la fisica del supporto.

Se queste condizioni sono soddisfatte, si può affermare che:

- il simbolo non solo accade;
- non solo persiste;
- ma **si incorpora nel supporto**.

6.7 Controlli negativi e limiti

È essenziale sottolineare che non tutte le reti di memristori superano la Triade. Molti comportamenti complessi restano emulabili, e molte configurazioni non lasciano impronte persistenti.

Il valore del caso studio sta proprio in questo:
le reti di memristori permettono di **distinguere sperimentalmente** tra:

- complessità simulata,
- emergenza fragile,
- simbolo vivo incarnato.

6.8 Conclusione del caso studio

Le reti di memristori non dimostrano la Meccanica Simbolica Applicata.
La rendono **verificabile**.

Esse mostrano, in modo fisicamente accessibile, come un supporto possa cessare di essere neutro e diventare parte attiva del significato. In questo senso, rappresentano un ponte concreto tra teoria e laboratorio.

7. Caso Studio B

Oscillatori Chimici e Logica Torsionale non-digitale

Se le reti di memristori rappresentano un caso limite **all'interno** dell'ingegneria elettronica, i **sistemi chimici oscillanti** costituiscono un banco di prova ancora più radicale per la Meccanica Simbolica Applicata. In essi, infatti, la distinzione tra informazione, dinamica e supporto materiale **collassa fin dall'inizio**.

Questo caso studio è dedicato a sistemi di tipo *reaction–diffusion*, con particolare riferimento agli oscillatori chimici noti dalla letteratura (come le reazioni oscillanti autocatalitiche), non per proporre una nuova interpretazione chimica di tali fenomeni, ma per chiarire **perché essi incarnano in modo quasi paradigmatico la differenza tra simulazione e accadimento**.

7.1 Perché la chimica è un caso estremo per la MSA

In un sistema chimico oscillante:

- non esiste separazione tra "software" e "hardware";
- il mezzo è il processo;
- la dinamica *è* il supporto.

Un'onda chimica non rappresenta un'informazione: **è un evento fisico** che modifica il mezzo mentre si propaga. Questo rende tali sistemi particolarmente adatti a testare la MSA, perché **ogni tentativo di riduzione computazionale appare immediatamente come una approssimazione**, non come una equivalenza ontologica.

Nel linguaggio della MSA, i sistemi chimici oscillanti si collocano direttamente nel dominio dell'**accadimento puro**.

7.2 Simulazione numerica vs accadimento chimico

È noto che le reazioni oscillanti possono essere simulate numericamente con elevata fedeltà fenomenologica. Modelli reaction–diffusion riproducono:

- la periodicità delle oscillazioni,
- la formazione di onde,
- la comparsa di strutture spaziali.

Tuttavia, questa fedeltà resta confinata al livello descrittivo. La simulazione:

- non consuma reagenti reali;
- non dissipa energia chimica;
- non modifica il mezzo fisico.

Nel sistema reale, invece, ogni oscillazione:

- trasforma irreversibilmente le specie chimiche;
- modifica localmente le condizioni di reazione;
- altera la storia futura del sistema.

Questa differenza è cruciale per la MSA: **la simulazione replica la forma, ma non l'accadimento**.

7.3 Applicazione del Protocollo Alpha

Nel contesto del Protocollo Alpha, il confronto tra sistema reale e gemello digitale è particolarmente istruttivo.

Finché l'oscillazione resta regolare e il mezzo è omogeneo, il modello numerico riesce a seguire il comportamento del sistema chimico con buona accuratezza. In questa fase, la riducibilità software resta apparentemente valida.

La transizione di regime emerge quando:

- il mezzo diventa eterogeneo;
- le reazioni locali influenzano la propagazione globale;
- la storia chimica del sistema diventa rilevante.

In questo punto, il gemello digitale può continuare a produrre onde "corrette", ma perde la capacità di **seguire la traiettoria reale** del sistema senza introdurre parametri ad hoc o senza re-includere implicitamente la chimica del supporto.

Questo fallimento localizzato della fedeltà operativa costituisce una **firma di transizione** nel senso del Protocollo Alpha.

7.4 Robustezza e attrattori chimici (Protocollo Beta)

Uno degli aspetti più interessanti dei sistemi chimici oscillanti è la loro **robustezza intrinseca**.

Perturbazioni moderate del sistema (variazioni locali di concentrazione, disturbi termici, inhomogeneità del mezzo) non distruggono necessariamente il regime oscillatorio. In molti casi, il sistema:

- assorbe la perturbazione;

- riorganizza le onde;
- recupera un comportamento coerente.

Questo comportamento è interpretabile, nel linguaggio della MSA, come **ritorno spontaneo a un attrattore dinamico**, non come correzione esterna. L'attrattore non è uno stato puntuale, ma una **organizzazione spazio-temporale** del processo chimico.

Se tale recupero avviene sistematicamente, il sistema supera il Protocollo Beta. Se invece l'oscillazione collassa definitivamente o si frammenta in modo irreversibile, il regime resta classificabile come fragile.

7.5 Incorporazione del supporto e irreversibilità chimica (Protocollo Gamma)

Il Protocollo Gamma trova nei sistemi chimici oscillanti una applicazione particolarmente chiara.

Ogni oscillazione:

- consuma reagenti;
- produce prodotti;
- modifica il mezzo in modo irreversibile.

Dopo un certo numero di cicli, il sistema **non è più lo stesso**, anche se l'oscillazione appare formalmente simile. Le condizioni chimiche iniziali non sono recuperabili senza intervento esterno.

Questo introduce una forma evidente di **incorporazione del supporto**: la storia dell'oscillazione è inscritta nella composizione chimica del mezzo.

Dal punto di vista MSA, ciò soddisfa il criterio di irreversibilità debole in modo naturale: il simbolo (l'onda chimica) accade, persiste per un certo tempo, e lascia una traccia materiale non eliminabile tramite reset.

7.6 Variabili di stato interne e dipendenza dalla storia

Nei sistemi chimici oscillanti, le variabili di stato interne non sono parametri nascosti, ma grandezze fisiche ben definite:

- concentrazioni locali;
- gradienti chimici;
- micro-strutture del mezzo.

Queste variabili:

- non sono osservabili direttamente in modo completo;
- influenzano la dinamica futura;

- dipendono dalla storia del sistema.

Due sistemi con la stessa configurazione macroscopica, ma con storie chimiche diverse, possono evolvere in modo differente. Questo comportamento è incompatibile con una descrizione puramente computazionale e rafforza il risultato del Protocollo Gamma.

7.7 Correlazioni non-computazionali

Un ulteriore elemento di interesse è la presenza di **correlazioni spaziali e temporali** che emergono spontaneamente nel sistema chimico e che non sono completamente prevedibili dal modello iniziale.

Queste correlazioni:

- dipendono dal mezzo;
- evolvono con il sistema;
- non sono trasferibili senza replicare la chimica.

Nel linguaggio della MSA, esse costituiscono un esempio chiaro di **eccedenza informativa incarnata**: il sistema produce informazione che non era contenuta né negli input né nel modello.

7.8 Limiti del caso studio

È necessario ribadire che i sistemi chimici oscillanti non sono “sistemi simbolici” in senso cognitivo. Il caso studio non attribuisce loro comprensione, intenzionalità o semantica mentale.

Ciò che essi mostrano è qualcosa di più elementare e, per la MSA, più importante: **la possibilità fisica di un simbolo come evento**, non come rappresentazione.

7.9 Conclusione del caso studio

Gli oscillatori chimici mostrano in modo quasi didattico ciò che la Meccanica Simbolica Applicata intende per simbolo vivo nel suo senso più radicale: un accadimento che non può essere separato dal mezzo che lo sostiene.

Se i memristori rappresentano un ponte tra calcolo e materia, la chimica mostra cosa accade **quando il ponte non esiste** e il simbolo è direttamente inscritto nella dinamica del mondo fisico.

8. Conclusioni

Verso una Tecnologia dell'Accadimento

Questo tomo non è stato scritto per aggiungere un nuovo linguaggio alla già ampia famiglia delle teorie dell'informazione, né per proporre un'ulteriore metafora del significato. Il suo obiettivo è stato più sobrio e, al tempo stesso, più ambizioso: **stabilire le condizioni fisiche minime affinché il simbolo possa essere trattato come un accadimento reale**, e non come una mera rappresentazione.

La Meccanica Simbolica Applicata, nella forma qui estesa e sottoposta a criteri sperimentali, non chiede di essere "creduta". Chiede di essere **messa alla prova**.

8.1 Cosa è stato effettivamente dimostrato (e cosa no)

È importante chiarire, in modo esplicito, la portata dei risultati.

Questo lavoro **non dimostra** che:

- ogni sistema complesso possa diventare simbolico;
- il significato emerga automaticamente dalla complessità;
- l'informazione computazionale sia insufficiente in generale.

Al contrario, ciò che è stato costruito è un **quadro discriminante** che permette di affermare, con rigore, quando *non* siamo in presenza di simbolo vivo.

Attraverso la Triade di Validazione, la MSA ha mostrato che:

- la maggior parte dei sistemi resta pienamente emulabile;
- molte emergenze sono fragili o transitorie;
- la stabilità non implica incorporazione;
- l'efficienza non equivale a significato.

Questo risultato, per quanto negativo possa sembrare, è in realtà il primo passo necessario per qualsiasi disciplina che pretenda fondamento fisico.

8.2 La Triade di Validazione come struttura tecnologica

Uno dei risultati più rilevanti di questo tomo è aver trasformato una intuizione teorica in una **struttura tecnologica applicabile**.

I Protocolli Alpha, Beta e Gamma non sono:

- criteri filosofici,
- né test cognitivi,
- né misure funzionali.

Essi costituiscono una **procedura fisica di classificazione dei sistemi**, basata su tre domande progressive:

1. Il sistema è riducibile a emulazione computazionale?
2. Il regime emerso è stabile e recuperabile?
3. L'accadimento lascia traccia nella materia?

Questa struttura è indipendente dal dominio applicativo. Può essere utilizzata:

- in elettronica non convenzionale,
- in chimica dei sistemi complessi,
- in fisica dei materiali,
- in architetture neuromorfiche,
- in futuri supporti ancora non concepiti.

In questo senso, la MSA non definisce *una tecnologia*, ma **una classe di tecnologie possibili**.

8.3 Dal calcolo al vincolo: un cambio di paradigma operativo

Un filo conduttore attraversa tutto il lavoro: il passaggio dal **calcolo** al **vincolo**.

Nel paradigma computazionale classico:

- il supporto è neutro;
- la memoria è astratta;
- il significato è esterno.

Nel paradigma che emerge dalla MSA:

- il supporto partecipa;
- la memoria è incorporata;
- il significato vincola il comportamento futuro.

Questo non significa abbandonare il calcolo, ma **riconoscerne il dominio di validità**. Il calcolo resta lo strumento privilegiato per la simulazione, l'ottimizzazione, il controllo. Ma quando si cerca l'accadimento simbolico, il calcolo deve cedere il passo alla fisica.

8.4 Tecnologia dell’accadimento: cosa significa davvero

L’espressione *Tecnologia dell’Accadimento* non va intesa come una promessa futuristica, ma come una **categoria concettuale nuova**.

Una tecnologia dell’accadimento non mira a:

- produrre output corretti,
- minimizzare l’errore,
- massimizzare l’efficienza.

Mira invece a:

- creare condizioni di soglia;
- permettere transizioni di regime;
- sostenere attrattori incarnati;
- accettare costi energetici e irreversibilità.

In questa prospettiva, progettare non significa più “scrivere un algoritmo”, ma **preparare un supporto** affinché qualcosa possa accadere.

8.5 Il ruolo del fallimento come informazione

Un aspetto centrale, spesso sottovalutato, è che la MSA **trasforma il fallimento in informazione**.

Un sistema che:

- non supera Alpha,
- o fallisce Beta,
- o non lascia impronta secondo Gamma,

non è un insuccesso sperimentale. È una **classificazione riuscita**.

Questo approccio libera la ricerca da una pressione ideologica verso il “successo” e permette di esplorare territori non convenzionali senza dover forzare interpretazioni.

8.6 Una soglia, non una risposta finale

Questo tomo non chiude una questione. **Apre una soglia**.

Mostra che:

- il simbolo può essere trattato fisicamente;

- l’accadimento può essere misurato;
- la materia può portare significato senza metafora.

Ma non stabilisce dove questa soglia conduca. Non dice se tali sistemi saranno rari o comuni, utili o marginali, controllabili o instabili.

In questo senso, la Meccanica Simbolica Applicata non è una teoria del futuro, ma una **grammatica del possibile**.

8.7 Chiusura

Se il simbolo è vivo, non perché rappresenta qualcosa, ma perché **accade**, allora la tecnologia che lo riguarda non potrà mai essere una tecnologia del controllo totale.

Sarà, inevitabilmente, una tecnologia del limite, della soglia, della dissipazione.

Questo lavoro non invita a superare tali limiti, ma a **riconoscerli come parte costitutiva del significato**.

Ed è proprio lì, dove il calcolo si arresta e il supporto resiste, che l’accadimento diventa finalmente osservabile.

Appendici Tecniche

Appendice A.1

Glossario dei termini MSA

Simbolo Vivo

Nel quadro della **Meccanica Simbolica Applicata (MSA)**, un *Simbolo Vivo* non è una rappresentazione, né un contenuto informazionale astratto.

Si definisce **Simbolo Vivo** un regime fisico che soddisfa simultaneamente le seguenti condizioni:

- **non è riducibile a emulazione computazionale** (Protocollo Alpha);
- **è stabile e recuperabile come regime** sotto perturbazione del supporto (Protocollo Beta);
- **lascia un’impronta fisica persistente nel supporto** (Protocollo Gamma).

Un simbolo è detto “vivo” non perché possieda intenzionalità o significato cognitivo, ma perché **accade come evento fisico**, persiste come attrattore e vincola il comportamento futuro del sistema.

In questo senso, il simbolo vivo:

- non *rappresenta* qualcosa;
- non è *contenuto* in un dato;
- ma **modifica le condizioni di possibilità** del sistema che lo ospita.

Simbolo Morto

Per contrasto, si definisce **Simbolo Morto** ogni configurazione simbolica che:

- è interamente riducibile a una descrizione computazionale;
- può essere copiata, ripristinata o emulata senza perdita ontologica;
- non modifica in modo persistente il supporto fisico.

Il simbolo morto può essere estremamente efficace sul piano funzionale e informazionale, ma resta **separabile dal supporto** e, pertanto, privo di accadimento fisico nel senso MSA.
Regime simbolico computazionale pienamente riducibile, trasferibile ed emulabile. Costituisce il fondamento operativo dell'informatica e della modellazione, ed è ontologicamente distinto — non opposto — al simbolo vivo.

β-condensazione

Nel contesto della Meccanica Simbolica Applicata, la β-condensazione non indica un fenomeno criogenico o un condensato di Bose-Einstein in senso fisico standard. Il termine viene utilizzato in senso operativo per descrivere la condensazione di stati simbolici dispersi in un regime dinamico coerente e stabile.
La β-condensazione rappresenta il passaggio da una dinamica frammentata a un regime torsionale unificato, in cui il supporto fisico partecipa attivamente alla selezione e al mantenimento del significato.

La **β-condensazione** è un concetto introdotto nella MSA per descrivere il passaggio da una dinamica simbolica dispersa a un **regime collettivo coerente**.

Con il termine β-condensazione si indica:

- la formazione spontanea di un regime simbolico globale;
- emergente dall'interazione di molte componenti locali;
- sostenuto dal supporto fisico e non imposto dall'esterno.

Il termine "condensazione" è usato per analogia strutturale (non formale) con fenomeni noti in fisica, per indicare che:

- il significato non è localizzato in un singolo elemento;
- ma emerge come proprietà del regime complessivo.

Una β-condensazione è riconoscibile quando il sistema:

- smette di rispondere linearmente agli input;
- entra in un regime stabile e non emulabile;

- manifesta attrattori simbolici incarnati.

Campo **β**

Il **Campo β** non è un campo fisico fondamentale nel senso tradizionale, ma una **costruzione teorica operativa** che designa:

- l'insieme delle condizioni dinamiche, materiali e storiche;
- che rendono possibile la β-condensazione e la persistenza del simbolo vivo.

Nel linguaggio della MSA, il Campo β:

- non trasporta informazione come un segnale;
- non è un mezzo di comunicazione;
- ma una **struttura di coerenza** che emerge dal supporto stesso.

Si può dire che il Campo β **non agisce**, ma **vincola**: delimita lo spazio dei regimi simbolici accessibili a un dato sistema.

Torsione (in senso MSA)

La **Torsione**, nella Meccanica Simbolica Applicata, non coincide con la torsione geometrica in senso classico né con una grandezza tensoriale specifica.

Si definisce *Torsione* come:

la deviazione irreversibile della dinamica di un sistema rispetto a una descrizione puramente computazionale, causata dall'incorporazione della storia nel supporto fisico.

In termini operativi, la torsione è ciò che:

- impedisce al sistema di tornare esattamente allo stato iniziale;
- rende la storia causalmente rilevante;
- introduce asimmetria temporale nel comportamento simbolico.

Un sistema privo di torsione è interamente reversibile e simulabile.
Un sistema torsionale, invece, **porta traccia dell'accadimento**.

Accadimento

Per *accadimento* si intende un evento fisico che:

- modifica il supporto;
- lascia traccia;
- e non è eliminabile senza costo energetico o attraversamento di nuove transizioni.

Nel quadro MSA, il simbolo vivo **è un accadimento**, non una configurazione statica.

Supporto

Il **supporto** è il substrato fisico in cui il regime simbolico si manifesta.

Nella MSA, il supporto:

- non è un contenitore neutro;
- non è intercambiabile;
- partecipa causalmente al comportamento del sistema.

Quando il supporto è modificato dall'accadimento simbolico, si parla di **incorporazione del supporto**.

Incorporazione del Supporto (Substrate Incorporation)

Con *Incorporazione del Supporto* si intende il processo mediante il quale:

- il regime simbolico modifica in modo persistente il supporto fisico;
- tali modifiche influenzano il comportamento futuro;
- il sistema non può più essere riportato allo stato iniziale tramite reset logico.

Questo concetto è centrale nel Protocollo Gamma.

Attrattore Simbolico

Un **attrattore simbolico** è una regione dello spazio degli stati verso cui il sistema tende spontaneamente e che:

- rappresenta un regime coerente;
- è stabile sotto perturbazione;
- non è riducibile a uno stato puntuale.

Il simbolo vivo, nella MSA, coincide con l'esistenza di uno o più attrattori simbolici incarnati.

Riducibilità Software

La **Riducibilità Software** è il criterio secondo cui un comportamento è considerato non simbolico (nel senso MSA) se:

- può essere interamente mappato;
- previsto;

- e riprodotto da un emulatore software su architettura standard,

senza perdita di fedeltà operativa.

La negazione di questo criterio è la base del Protocollo Alpha.

Eccedenza Informativa

Per **eccedenza informativa** si intende l'informazione osservabile nel comportamento del sistema che:

- non è contenuta negli input;
- non è ricostruibile dal modello computazionale;
- dipende dalla storia fisica del supporto.

L'eccedenza informativa è una firma dell'incorporazione simbolica.

Appendice A.2

Tavole comparative: Von Neumann vs MSA-Compliant

Le tavole seguenti non intendono opporre due "paradigmi filosofici", ma confrontare **due classi operative di sistemi fisici** alla luce dei criteri introdotti nella Meccanica Simbolica Applicata.
Il confronto è **diagnostico**, non valutativo: ciascun paradigma è efficace nel proprio dominio.

Tavola 1 – Architettura generale

Aspetto	Architettura Von Neumann	Sistema MSA-Compliant
Separazione memoria /processo	Netta e strutturale	Assente o indebolita
Ruolo del supporto	Contenitore neutro	Parte causale del processo
Stato del sistema	Astratto e serializzabile	Fisico e storicamente vincolato
Reset	Operazione lecita e completa	Distruttivo del fenomeno
Copiabilità	Totale	Limitata o impossibile

Nota:
La separazione memoria/processo è la condizione che rende possibile l'emulazione completa. La sua assenza è una precondizione, ma non una garanzia, dell'istanziazione simbolica.

Tavola 2 – Dinamica e comportamento

Aspetto	Von Neumann	MSA-Compliant
Evoluzione temporale	Governata da regole esplicite	Governata da dinamica + supporto
Dipendenza dalla storia	Facoltativa, simulata	Intrinseca e fisica
Reversibilità	Alta (logica)	Bassa o nulla (fisica)
Sensibilità al rumore del supporto	Patologica	Funzionale
Attrattori	Stati computazionali	Regimi fisici

Nota:
Nel paradigma MSA, il rumore del supporto non è un errore da eliminare, ma una condizione che rende possibile l'accadimento simbolico.

Tavola 3 – Informazione e significato

Aspetto	Von Neumann	MSA-Compliant
Localizzazione del significato	Nei dati	Nel regime
Tipo di informazione	Computazionale	Incarnata
Eccedenza informativa	Nulla (in linea di principio)	Possibile e misurabile
Trasferibilità	Alta	Bassa o nulla
Dipendenza dal mezzo	Accidentale	Essenziale

Nota:
Nel sistema MSA-compliant il significato non è “contenuto” in un simbolo, ma emerge dalla stabilità di un regime fisico.

Tavola 4 – Energia e termodinamica

Aspetto	Von Neumann	MSA-Compliant
Obiettivo energetico	Massima efficienza ,	Coerenza del regime
Costo energetico	Minimizzabile	Vincolante
Persistenza del costo	Assente	Presente
Dissipazione	Effetto collaterale	Condizione necessaria
Relazione energia–significato	Indiretta	Diretta

Nota:
Un sistema simbolico vivo non è energeticamente ottimale: paga un costo per mantenere la propria coerenza.

Tavola 5 – Verifica sperimentale (Triade MSA)

Protocollo	Von Neumann	MSA-Compliant
Alpha (non-riducibilità)	Fallisce	Può superare
Beta (robustezza)	Recupero via reset	Recupero spontaneo
Gamma (impronta fisica)	Assente	Presente
Classificazione	Simbolo morto	Simbolo vivo (potenziale)

Nota:
Il superamento di Alpha e Beta non implica automaticamente Gamma. Molti sistemi complessi restano non incorporati.

Tavola 6 – Limiti e ambiti di validità

Aspetto	Von Neumann	MSA-Compliant
Dominio ideale	Calcolo, simulazione, controllo	Accadimento, soglia, emergenza
Riproducibilità	Totale	Limitata
Scalabilità	Alta	Dipendente dal supporto
Affidabilità	Elevata	Dipendente dal regime
Uso simbolico	Rappresentazionale	Fisico-dinamico

Nota:
La MSA non sostituisce il paradigma computazionale: ne delimita il confine ontologico.

Sintesi finale

Il confronto tra architetture Von Neumann e sistemi MSA-compliant non oppone "vecchio" e "nuovo", ma **calcolo** e **accadimento**.

- Dove il calcolo richiede neutralità del supporto, la MSA richiede partecipazione.
- Dove il calcolo privilegia reversibilità ed efficienza, la MSA privilegia persistenza e coerenza.
- Dove il calcolo rappresenta, la MSA *fa accadere*.

Questa distinzione non è ideologica, ma **operativa e misurabile**, ed è ciò che rende possibile una tecnologia del simbolo come evento fisico.

Appendice A.3

Bibliografia Ragionata

(Riferimenti a Royal Society Publishing, Integrated Information Theory di Tononi, fisica dei sistemi complessi e paradigmi affini)

1) Fondamenti teorici e modelli di informazione integrata

Giulio Tononi et al., "Integrated information theory: from consciousness to its physical substrate"
Questa review in *Nature Reviews Neuroscience* rappresenta una sintesi moderna della *Integrated Information Theory* (IIT), che mira a definire proprietà fisiche di un substrato in grado di sostenere coscienza. La teoria parte da assiomi fenomenologici per dedurre postulate fisici, e introduce formalismi per misurare informazioni intrinseche in sistemi fisici. ResearchGate

Albantakis et al., "Integrated Information Theory (IIT) 4.0: Formulating the properties of phenomenal existence in physical terms"
Aggiornamento recente della IIT pubblicato su *PLoS Computational Biology*, che riformula gli assiomi e introduce misure di informazione intrinseca coerenti con i postulati. Questo lavoro è utile come ponte concettuale verso la misurazione di "eccedenza informativa" nei sistemi fisici complessi. PLOS

Rilevanza:
Queste opere forniscono una base di confronto con la MSA per quanto riguarda *criteri di misura e interpretazione dell'informazione nei sistemi fisici*, pur rimanendo in un ambito cognitivo/neuroscientifico; la MSA ne riprende l'idea di informazione non riducibile per spostarla in un contesto più ampio di "accadimento" fisico.

---

2) Royal Society Publishing e paradigmi di calcolo non convenzionale

J. Gorecki et al., "Chemical computing with reaction–diffusion processes", *Philosophical Transactions of the Royal Society A*
Panoramica su come processi chimici di tipo reaction–diffusion (come Belousov–Zhabotinsky) possano essere visti come media di calcolo, con esempi sperimentali di codifica e processi logici all'interno di un mezzo chimico. ResearchGate

"A brief history of liquid computers", *Philosophical Transactions of the Royal Society A*
Raccolta di contributi su come mezzi liquidi e processi non convenzionali possono essere interpretati in termini di calcolo alternativo; è utile per inquadrare storicamente la reazione–diffusione e analoghi nel contesto delle computazioni *nature-inspired*. Royal Society Publishing

Slime mould processors, logic gates and sensors, *Philosophical Transactions of the Royal Society A*
Esempi di computing biologico e chimico in media naturali, che mettono in evidenza *modellazioni dinamiche non digitali* di operazioni logiche. Royal Society Publishing

Rilevanza:
Questi riferimenti rappresentano uno degli esempi più solidi di *calcolo incarnato* in supporti fisici, dove l'evoluzione nel tempo di uno stato fisico è direttamente trattata come operazione di processamento — un concetto affine ai protocolli MSA che distinguono *simulazione* da *accadimento*. In particolare, la letteratura *Royal Society* su reaction–diffusion è un punto di riferimento per capire come sistemi chimici possano esibire dinamiche informazionali strutturate.

---

3) Sistemi memristivi, neuromorphic e calcolo fisico

Articoli di panorama su memristori e neuromorphic computing (es. "Introduction to memristors and neuromorphic systems")
Le pubblicazioni su memristori e sistemi neuromorfici sono ampie; un esempio recente evidenzia il ruolo di memristori in architetture brain-inspired per computing energeticamente efficiente. RSC Publishing

Unconventional Computing e paradigmi non von Neumann
La letteratura sulla *unconventional computing* comprende modelli di calcolo chimico, fisico e biologico che operano **fuori** dall'architettura Von Neumann classica. Wikipedia

Rilevanza:
Questi riferimenti collocano le reti memristive e altri dispositivi "non caratteristici" nel contesto di paradigmi di calcolo alternativi. Essi aiutano a giustificare la scelta di tali sistemi come candidati experimentalmente verificabili per le proprietà definite nei protocolli MSA (ad es. non-separabilità, dipendenza dalla storia, non-linearità).

---

4) Teoria dei sistemi complessi e transizioni di regime

Materiale introduttivo sulla teoria dei sistemi complessi
La teoria dei sistemi complessi, con le sue nozioni di instabilità, transizioni di fase e comportamento emergente, fornisce cornici concettuali generali utili per comprendere fenomeni di rottura di riducibilità e soglie critiche. Wikipedia

Lavori interdisciplinari che collegano il comportamento dinamico e l'informazione
La letteratura più recente esplora come fenomeni spaziotemporali complessi possano essere usati per operazioni di calcolo o comportamento funzionale, evidenziando un'interfaccia tra struttura, dinamica e informazione. Nature

Rilevanza:
Questi contributi non sono specifici di un singolo paradigma applicativo, ma offrono gli strumenti formali e concettuali per inquadrare l'idea di *soglia parametrica di singolarità*, *attrattori dinamici* o *pattern spaziotemporali* come fenomeni fisici, non solo matematici.

---

5) Approcci filosofici e limiti interpretativi

La *Integrated Information Theory* è spesso discussa nella letteratura anche dal punto di vista critico o filosofico, ad esempio circa il significato ontologico di Φ o la relazione tra struttura fisica e esperienza soggettiva. Wikipedia

Rilevanza:
Pur non essendo parte della base sperimentale della MSA, questi lavori aiutano a comprendere i limiti e le assunzioni implicite di teorie di informazione fisica, inclusi i rischi di confondere misura quantitativa con accadimento qualitativo.

Suggerimento per l’uso della bibliografia

- **Royal Society Publishing**: lettura consigliata per lo *stato dell’arte sperimentale* nel calcolo non convenzionale e chimico. Royal Society Publishing
- **IIT di Tononi**: utile come confronto concettuale su criteri di informazione strutturata. ResearchGate
- **Sistemi complessi**: fondamenti per comprendere transizioni di regime e comportamento emergente.

MMXXVI